

Monika Kolvenbach / Andreas Lötscher /
Hans Dieter Lutz (Hrsgg.)

Künstliche Intelligenz und natürliche Sprache

Sprachverstehen und Problemlösen mit dem Computer

**FORSCHUNGSBERICHTE DES
INSTITUTS FÜR DEUTSCHE SPRACHE
MANNHEIM**

herausgegeben von
Ulrich Engel und Gerhard Stickel
Schriftleitung: Eva Teubert

Band 42

**MONIKA KOLVENBACH/
ANDREAS LÖTSCHER/
HANS DIETER LUTZ (Hrsgg.)**

Künstliche Intelligenz und natürliche Sprache

**Sprachverstehen und Problemlösen
mit dem Computer**



Gunter Narr Verlag · Tübingen

CIP-Kurztitelaufnahme der Deutschen Bibliothek

Künstliche Intelligenz und natürliche Sprache:

Sprachverstehen u. Problemlösen mit d. Computer / Monika Kolvenbach . . .
(Hrsgg.). — Tübingen: Narr, 1979.

(Forschungsberichte / Institut für Deutsche Sprache Mannheim; Bd. 42)

ISBN 3 - 87808 - 642 - 3

NE: Kolvenbach, Monika [Hrsg.]

© 1979 · Gunter Narr Verlag Tübingen

Alle Rechte vorbehalten. Nachdruck oder Vervielfältigung, auch
auszugsweise, in allen Formen wie Mikrofilm, Xerographie, Mikrofiche,
Mikrocard, Offset verboten.

Druck: Müller + Bass, Tübingen

Printed in Germany

ISBN 3 - 87808 - 642 - 3

INHALT

Vorwort	7
Hanno Wulz/Gisela Zifonun	
Automatische Problemlösung und Sprachverarbeitung als Forschungsgegenstände	15
G.L. Berry-Rogghe/Hans Dieter Lutz/Kaija Saukko	
Das Informationssystem PLIDIS	39
Gisela Zifonun	
Formale Repräsentation natürlichsprachlicher Äußerungen	93
Andreas Lötscher	
Automatische syntaktische Analyse des Deutschen mit Übergangsnetzwerken	135
Andreas Lötscher/Monika Kolvenbach	
Morphosyntaktische Analyse in einem Frage-Antwort-System	187
Hanno Wulz	
Aspekte der automatischen Überführung natürlichsprachlicher Formulierungen in eine formale Repräsentationssprache	227
Werner Dilger	
Theorembeweiser als Bestandteil der Problemlösungskomponente	265
G.L. Berry-Rogghe/Werner Dilger	
Konzeption eines Termininterpreters	289
Rolf Guntermann	
Datenbasisverwaltung in einem Frage-Antwort-System	305
Literaturverzeichnis	315
Register	323

VORWORT

Der Bürger einer hochentwickelten Industriegesellschaft wird ständig mit dem Computer konfrontiert: Elektronische Datenverarbeitung wird bei der Buchhaltung selbst kleinerer Betriebe, bei der Kontenführung der Sparkassen und Banken, bei der Gebührenrechnung von Kommunen und Behörden eingesetzt, natürlich auch beim Aufbau und Einsatz der verschiedenen Informationssysteme, die unserer Zeit das Gepräge geben.

Der breiten Öffentlichkeit weniger bekannt sind Problemlösungsverfahren mit Hilfe des Computers (wie Managementinformationssysteme bzw. Planungs- und Entscheidungssysteme) und die Versuche, mit Hilfe der elektronischen Datenverarbeitung Kommunikationsprozesse zu rationalisieren, zu standardisieren und vor allem kostengünstiger zu gestalten; am bekanntesten sind dabei noch die Versuche der automatischen Übersetzung, die in den 50er und 60er Jahren die Diskussion beherrschten.

Es ist wohl unbestreitbar, daß die elektronische Datenverarbeitung eine große Zahl von Erleichterungen mit sich bringt. Dafür wird immer wieder ins Feld geführt, daß die übersichtliche Speicherung riesiger Datenmengen, der schnelle Zugriff, die exakt festgelegten Abfragemöglichkeiten und vieles andere mehr Möglichkeiten der Informationsvermittlung böten, von denen man früher nur zu träumen wagte. Daß das elektronische Zeitalter auch neuartige Probleme mit sich bringt, dringt erst langsam ins Bewußtsein. Am ehesten sorgt man sich noch um die Gefahr totaler Überwachung des Bürgers, die die mindestens in der Tendenz totale Datenspeicherung mit sich führen könnte, und um die Gefahr, daß mit der Einführung neuer elektronischer Technologien Arbeitsplätze wegrationalisiert werden und eine Wandlung in der Struktur tradierter Berufsbilder eintritt, auf die man nicht bzw. nicht genügend vorbereitet ist. Daneben treten andere, nicht minder wichtige Probleme in den Hintergrund. Eines betrifft das Verhältnis des Benutzers zur elektronischen Rechanlage, näm-

lich die Art der Kommunikation zwischen Mensch und Computer, auf die kurze Formel gebracht: den Mensch-Maschine-Dialog.

Ein großer Teil der Ängste, die Außenstehende angesichts der zunehmenden Bedeutung der elektronischen Datenverarbeitung befallen, rührt ja daher, daß man Konstruktion und Verfahrensweise des Computers, dessen man sich bedienen muß, nicht durchschaut, ja daß man gezwungen ist, sich generell einer Vermittlungsinstanz zu bedienen, nämlich des Programmierers, der allein in der Lage ist, mit der Maschine in unmittelbare Kommunikation zu treten. Es geht dann weitgehend darum, daß der Anwender seine Probleme so formuliert, daß sie dem Programmierer einleuchten; der Programmierer wiederum muß in der Lage sein, sie in computergemäße Form zu übertragen. Bei diesem Übermittlungsprozeß sind Pannen, Verständigungsschwierigkeiten nicht ausgeschlossen. Der Anwender fühlt sich vielfach dem Programmierer ausgeliefert, der zwar Software-Experte, aber in der Regel nicht Experte für den Problembereich des Anwenders ist. Hinzu kommt in sehr vielen Fällen, daß die Programmierkapazität und die technischen Hilfsmittel nicht ausreichen, um die jeweiligen Anwenderprobleme adäquat in eine Computersprache zu übersetzen und so die Problemlösung mit Hilfe des Computers in dem gewünschten Sinne herbeizuführen.

Außerdem gehen mit der Einführung der elektronischen Datenverarbeitung vor allem im Bereich der Verwaltung neue Erscheinungsformen der Arbeitsteilung einher, deren schlimmste Konsequenz mit dem Schlagwort "entfremdete Arbeit" gekennzeichnet werden kann. Aufgrund der Aufspaltung eines Vorgangs in kleinste Teilschritte, für deren Bearbeitung z.B. ganz verschiedene Sachbearbeiter zuständig sind, hat der einzelne Sachbearbeiter keinen Überblick mehr über die Gesamtheit der Teilschritte und den Stellenwert der einzelnen Schritte. Da er zudem den Sinn dieser Aufspaltung nicht zu beurteilen vermag, nimmt sein Vermögen ab, sich mit der von ihm für einen Teilschritt geleisteten Arbeit voll zu identifizieren. Ausflüsse dieser Tendenz sind dann z.B. Auskünfte eines Sachbearbeiters des Inhalts, daß er einen aufgetretenen Fehler nicht feststellen könne, da das alles jetzt ja der Computer mache.

Aus dieser Situation ging das Bestreben hervor, Verfahren zu entwickeln, die eine unmittelbare Kommunikation zwischen Anwender und Computer ermöglichen. Konkret gesprochen: Der Anwender soll in die Lage versetzt werden, in seiner eigenen Sprache (z.B.: auf deutsch) dem Computer sein Problem zu vermitteln und damit zusammenhängende Fragen zu stellen. Dementsprechend muß der Computer in die Lage versetzt werden, diese Fragen zu "verstehen", sie in computerinterne Sprache umzusetzen, sie der Problemstellung entsprechend zu verarbeiten, Antworten in computerinterner Sprache zu formulieren und diese wiederum in natürliche Sprache (z.B.: deutsch) umzusetzen. Der Anwender stellt also seine Fragen auf deutsch und erhält die entsprechenden Antworten wiederum in deutscher Sprache formuliert. Es bedarf keiner zwischengeschalteten dritten Instanz mehr.

Diese Vorstellung erfordert ein Programmsystem, das den Computer entscheidend an der Problemlösung beteiligt und natürlich-sprachliche Ein- und Ausgabe ermöglicht. Man spricht in diesem Zusammenhang dann von natürlichsprachlich fundierten Dialogsystemen. Die Grundgedanken dazu sind von GEROLD UNGEHEUER seit 1970 entwickelt worden.

Der Vorteil eines solchen Verfahrens ist offensichtlich. Liegt ein entsprechendes Programmsystem vor, so ist der Benutzer nicht mehr auf den Programmierer angewiesen, mögliche Mißverständnisse bei der Informationsübermittlung vom Benutzer zum Programmierer werden ausgeschaltet. Damit ist nicht gesagt, daß politische und gesellschaftliche Probleme, die mit dem Einsatz der elektronischen Datenverarbeitung, speziell: großer Datenbanken und "intelligenten" Textverarbeitungssysteme, überhaupt entstehen, aus der Welt geschafft wären. Es ist aber ein Ansatzpunkt genannt, von dem her eine Chance besteht, Erscheinungsformen entfremdeter Arbeit und bestimmter Strukturen der Arbeitsteilung im Bereich des Einsatzes der elektronischen Datenverarbeitung schrittweise abzubauen bzw. rückgängig zu machen; denn gerade im Bereich der elektronischen Datenverarbeitung wird die exzessive Handhabung der Arbeitsteiligkeit mit dem Fehlen "menschengerechter" Lösungswege gerechtfertigt und "der" Computer zum Sündenbock stilisiert.

Der unmittelbare Mensch-Maschine-Dialog auf der Grundlage eines natürlichsprachlich fundierten Verfahrens erfordert die Konstruktion eines Frage-Antwort-Systems, das die natürliche Sprache nicht nur hinsichtlich ihrer Oberflächenerscheinungen, sondern auch hinsichtlich ihrer semantischen Eigenschaften und ihrer pragmatischen Verwendungszwecke zu verarbeiten vermag. Genau solche Frage-Antwort-Systeme stellen aber ein zentrales Problem auch innerhalb des Forschungsbereichs über sogenannte "Künstliche Intelligenz" ("artificial intelligence") dar. In diesem Forschungszweig sind nicht nur Grundfragen der Mensch-Maschine-Kommunikation ihrer Lösung nähergebracht worden, sondern es sind auch Muster für interdisziplinäre Arbeit gefunden worden. Die einzelnen Teilwissenschaften haben je ihre eigenen Zielsetzungen, Methoden und Objektvorstellungen. Beharren sie in sich selbst, so können sie nur isoliert nebeneinanderstehende Teillösungen erbringen. In einem funktionierenden Frage-Antwort-System müssen indessen Linguisten, Logiker und Informatiker eng zusammenarbeiten. Das hat andererseits zur Folge, daß keine der drei Teilwissenschaften - Linguistik, Logik und Informatik - in "reiner" Ausprägung zur Geltung kommt, und selbstverständlich darf keine der drei Teilwissenschaften einen Vorrang gegenüber den anderen beanspruchen. Dies wiederum bedeutet, daß z.B. linguistische Fragen teilweise auf eine Art gestellt und behandelt werden müssen, die ungewohnt erscheinen müssen innerhalb einer Linguistik, die sich nicht primär mit dem Problem von Frage-Antwort-Systemen beschäftigt.

Der Forschungsgegenstand "natürlichsprachliche Frage-Antwort-Systeme" kann mehrfachen Nutzen bringen. Die Unmittelbarkeit der Mensch-Maschine-Kommunikation, die Ausschaltung von im menschlichen Auffassungsvermögen liegenden Mißverständnissen wurde schon erwähnt. Hinzu kommt aber, daß durch solche Forschungen die Möglichkeit eröffnet wird, wichtige Einsichten in sprachliches Handeln und dessen Nachbildung zu gewinnen, und zwar in expliziten Modellen, die außerdem mit Hilfe von Computersystemen nachprüfbar gemacht werden. Dabei kann der Benutzer eines solchen Frage-Antwort-Systems nur sprachlich handeln. Er hat also keine Möglichkeit, die ihm sonst im alltäglichen Handeln noch zur Ver-

fügung stehenden Mittel wie Körpersprache, Gestik usw. in die Kommunikation einzubringen. Damit ergibt sich, verglichen mit alltäglicher Kommunikation, eine Art von Laborsituation, in der bestimmte andere Faktoren nicht berücksichtigt sind. Man mag dies als Manko ansehen. Trotzdem bleibt der unbestreitbare Vorteil, sprachliches Handeln quasi experimentell zu betrachten und zu analysieren und die dafür notwendigen Instrumente zu entwickeln.

Das Institut für deutsche Sprache beschäftigt sich seit Anfang der siebziger Jahre mit den Vorarbeiten zu einem solchen Informationssystem. 1971 begann man, die Möglichkeiten des Einsatzes der Datenverarbeitung zur Informationsgewinnung auf der Basis natürlicher sprachlicher Interaktion zu prüfen, dies allerdings zunächst eher theoretisch, und die Überprüfung wurde auch nur in Teilbereichen mit dem Rechner vorgenommen. Auskunft über einen Teil dieser Arbeiten geben die Bände 18 und 19 der Forschungsberichte des Instituts für deutsche Sprache, die über einen Versuch automatischer Syntaxanalyse der Arbeitsgruppe MasA berichten. Diese frühen Überlegungen wurden im Verlauf der Projekte Linguistische Datenverarbeitung (LDV) I und II (1971-1975) unter der Leitung von G. UNGEHEUER dahingehend konkretisiert, daß schließlich das experimentelle Frage-Antwort-System ISLIB (Informationssystem auf linguistischer Basis) erstellt wurde. ISLIB wurde auf kleine Sprachausschnitte des Deutschen und auf fallstudienhaft eingegrenzte Weltausschnitte angewandt: Börse, Probleme der Abfallbeseitigung. Dabei wurden zunächst noch verschiedene rivalisierende Konzepte zur Verarbeitung natürlicher Sprache, zur Repräsentation des Wissens und zur Fragebeantwortung durchgespielt. Das System war modular aufgebaut, so daß durch Austausch von Systemkomponenten die Möglichkeit zum Testen alternativer Ansätze gegeben war. Besonders im Bereich der linguistischen Analyse konnten ausbaufähige Ergebnisse erzielt werden.

Im Anschlußprojekt LDV III (seit 1976) wurde unter Leitung von D. KRALLMANN der Problemlösungsaspekt stärker in den Vordergrund gestellt; Arbeitsziel war das System PLIDIS (problemlösendes Informationssystem mit Deutsch als Interaktionsssprache). Dabei ergab sich durch die Zusammenarbeit mit einem künftigen Benutzer des

Systems, dem Ernährungsministerium des Landes Baden-Württemberg, ein stärkerer Anwendungsbezug für die Forschungsarbeiten.

Das System PLIDIS hat damit die Eigenschaft, sprachliche Äußerungen analysieren und in einem bestimmten Rahmen so weiterverarbeiten zu können, daß es sinnvolle Reaktionen auf diese Äußerungen liefern kann. Daß PLIDIS, so wie es bislang realisiert ist, schon ein für diesen Zweck voll ausgebautes und voll leistungsfähiges System darstelle, soll damit nicht behauptet werden.

Die Beiträge zu diesem Band stammen von Mitarbeitern am Projekt PLIDIS, das 1976 bis 1977 in der Abteilung Linguistische Datenverarbeitung des Instituts für deutsche Sprache durchgeführt wurde. Die einzelnen Aufsätze sind zwar durchaus im Zusammenhang mit den Projektarbeiten zu sehen: die Beschreibung des Systems PLIDIS, seine Konzeption und Realisierung von BERRY-ROGGHE et al. macht deutlich, wie die verschiedenen Problemaspekte mit der gemeinsamen Systemkonzeption von PLIDIS verbunden sind. Insgesamt handelt es sich jedoch nicht um eine Projektdokumentation im strengen Sinne, sondern um die eigenständige Aufarbeitung und Reflektion der im Projektziel angelegten prinzipiellen Fragestellungen.

Es hat sich als schwierig erwiesen, die einzelnen Beiträge so anzuordnen und aufeinander abzustimmen, daß damit alle möglichen Leserinteressen gleichermaßen berücksichtigt sind.

Unstrittig war, daß eine Hinführung zu der Problematik natürlicher sprachlicher Frage-Antwort-Systeme an den Anfang des Bandes zu stellen war. Der Beitrag von WULZ/ZIFONUN gibt nicht nur einen (wenn auch gestrafften) Überblick über den Forschungsbereich der künstlichen Intelligenz, sondern unternimmt auch den Versuch, den Forschungsansatz, der der Realisierung von PLIDIS zu Grunde liegt, aus der Problematik der künstlichen Intelligenz heraus plausibel zu machen.

Als zweiter Beitrag folgt eine Beschreibung des Systems PLIDIS von BERRY-ROGGHE et al. Damit soll dem Leser die Möglichkeit gegeben werden, unmittelbar zu verfolgen, wie die im ersten Beitrag

skizzierten Probleme der sprachorientierten Forschung über künstliche Intelligenz - Verarbeitung natürlicher Sprache, interne Repräsentation und Inferenzbildung - bei der Realisierung von PLIDIS angegangen worden sind. Es versteht sich, daß dies immer mit Bezug auf den exemplarisch ausgewählten Anwendungsbereich geschieht.

Unter den folgenden Beiträgen nimmt der von ZIFONUN wiederum eine zentrale Stellung ein, vor allem, weil die Frage der internen Repräsentation natürlichsprachlicher Formulierungen im Zentrum jeder Beschäftigung mit Frage-Antwort-Systemen steht. Denn die Verarbeitungen natürlichsprachlicher Formulierungen muß formale Ausdrücke als Ergebnis liefern, und diese Ausdrücke müssen den Bedingungen der internen Repräsentation genügen. Sie bilden auf der anderen Seite den Ausgangspunkt für die Inferenzbildung, mit deren Hilfe das Frage-Antwort-System sinnvoll auf die Bedürfnisse eines Benutzers reagieren kann.

Die Beiträge von LÖTSCHER, LÖTSCHER/KOLVENBACH und WULZ behandeln Probleme, die sich bei der Verarbeitung natürlichsprachlicher Formulierungen ergeben.

Die Beiträge von DILGER, BERRY-ROGGHE/DILGER und GUNTERMANN sind Fragestellungen aus dem Bereich der Inferenzbildung (und der Bereitstellung der dazu nötigen Daten) gewidmet.

Die letztgenannten sieben Beiträge ergänzen wiederum die Beschreibung des Informationssystems PLIDIS und vertiefen Teilaspekte, die dort nur summarisch behandelt werden konnten.

Verbindungen zwischen den einzelnen Beiträgen werden durch Querverweise und durch ein Register am Ende des Bandes gezogen.

Dem Leser mag noch folgender Hinweis nützlich sein: Die Beiträge von LÖTSCHER, LÖTSCHER/KOLVENBACH und WULZ wenden sich eher an einen Leserkreis mit Interesse an der Formalisierung morphosyntaktischer und tiefsyntaktischer Probleme. Die Aufsätze von DILGER, BERRY-ROGGHE/DILGER und GUNTERMANN konzentrieren sich auf Aspekte der logischen und logisch-pragmatischen Seite der Sprachverarbeitung und Frage-Beantwortung.

(Abschluß der Manuskripte: Juni bzw. Oktober 1978)

Hanno Wulz

Gisela Zifonun

AUTOMATISCHE PROBLEMLÖSUNG UND SPRACHVERARBEITUNG ALS FORSCHUNGSGEGENSTÄNDE

1. Der Forschungsbereich "Künstliche Intelligenz"
 - 1.1. Forschungsgeschichtlicher Hintergrund
 - 1.2. Forschungsschwerpunkte
 - 1.3. Zentren der KI-Forschung
2. Problemlösung als gemeinsamer Forschungsinhalt
 - 2.1. Zum Problemlösungsbegriff
 - 2.2. Das Repräsentationsproblem
 - 2.3. Lösungsstrategien
3. Sprachorientierte KI-Forschung
 - 3.1. Problemlösung im Bereich sprachorientierter KI-Forschung
 - 3.2. Grundsätzliche Probleme
 - 3.2.1. Repräsentation des Inhalts natürlichsprachlicher Formulierungen
 - 3.2.2. Inferenzregeln und Inferenzmechanismen
 - 3.2.3. Parsing und Überführung in eine interne Repräsentation
 - 3.2.4. Generierung natürlichsprachlicher Formulierungen
4. Sprachorientierte KI-Forschung und Linguistik
 - 4.1. Abgrenzung der sprachorientierten KI-Forschung von Anwendungen der EDV in der Linguistik
 - 4.2. Der Beitrag der sprachorientierten KI-Forschung zur Erforschung der Sprache

Anmerkungen

1. Der Forschungsbereich "Künstliche Intelligenz"

1.1. Forschungsgeschichtlicher Hintergrund

Die sich in der Zeit nach dem zweiten Weltkrieg abzeichnende Verfügbarkeit programmierbarer Rechenautomaten ermöglichte neue Spekulationen über die alte Frage, inwieweit "Denkmaschinen" konstruierbar sind, denen das Merkmal "intelligent" zugebilligt werden kann. Auf diese Frage zielen sowohl die Überlegungen von TURING, 1950, über 'Computing Machinery and Intelligence' als auch die Diskussion von SHANNON, 1950, inwieweit es möglich sei, eine Maschine so zu programmieren, daß sie Schach spielen kann. Diese Fragestellung wurde konstituierend für einen Forschungsbereich, der sich die Untersuchung "künstlicher Intelligenz"¹ (KI) zur Aufgabe stellt mit dem Ziel, Maschinen zu konstruieren bzw. Computerprogramme zu entwickeln, die Aufgaben erfüllen können, die normalerweise menschliche Intelligenz und intellektuelle Anstrengungen erfordern (NILSSON, 1971). Als Beispiele solcher intelligenter Aufgaben findet man in der KI-Forschung vor allem: Puzzles, (Brett-)Spiele, Denksportprobleme und einfache Induktionsaufgaben.

Die Anfangsphase der KI-Forschung in den fünfziger Jahren ist gekennzeichnet durch Systeme, die Puzzles und einfache logische Probleme lösen. Am einflußreichsten wurden "The Logic Theorist" (NEWELL/SIMON, 1956) und "General Problem Solver" (GPS: NEWELL/SHAW/SIMON, 1960), da in ihnen bereits wichtige Konzepte der KI, wie Problemlösung und Lösungsstrategie, vorstrukturiert wurden. Aufbauend auf SHANNON's Aufsatz wurden ab 1957 zunächst einfachere Schachspiel-Programme entwickelt, denen Ende der sechziger Jahre Versionen folgten, die in der Strategie der Spielführung und der Vorausplanung von Spielzügen sich mit guten Schachspielern messen ließen. Daneben stehen Entwicklungen von Programmen, die Poker, Bridge u.a. spielen. Ergänzung erhielt der KI-Zweig "Game Playing" durch die theoretische Analyse von Spielstrategien.

Die Notwendigkeit, komplexe Regularitäten etwa eines Spiels oder strukturelle Zusammenhänge von Einzeldaten etwa in Form von Strukturbäumen darzustellen, führte zur Entwicklung von Programmier-

sprachen, die im Gegensatz zu maschinennahen Sprachen nicht an Ablaufstrukturen und Verarbeitungsdetails eines Computers, sondern an den Darstellungsnotwendigkeiten der zu lösenden logisch-mathematischen Probleme orientiert sind. Die Entwicklung von KI-adäquaten Programmiersprachen erhielt einen besonderen Impuls durch die Konstruktion der Programmiersprache LISP (McCARTHY, 1960), auf der später noch abstraktere "very high level"-Programmiersprachen aufbauten.

Ab Anfang der sechziger Jahre entstanden dann Systeme zum Beweisen von mathematischen und logischen Sätzen, die durch die Arbeit von ROBINSON, 1965 besonderen Auftrieb erhielten: Er entwickelte ein Verfahren zum Beweisen von Sätzen der Prädikatenlogik erster Stufe, das auf nur einer Ableitungsregel, dem sogenannten Resolution-Prinzip, beruht. Dieses Verfahren konnte auf verhältnismäßig einfache Weise von Computerprogrammen automatisch realisiert werden.

Ein naheliegender Gegenstand für die KI-Forschung war die Beschäftigung mit der natürlichen Sprache bzw. mit einzelnen natürlichen Sprachen, da die sprachliche Kommunikation eine der grundlegendsten Formen intelligenten Verhaltens darstellt. Die Nachahmung dieser Fähigkeit in Computersystemen ist deshalb ein besonderer Prüfstein dafür, inwieweit künstliche Intelligenz realisiert werden kann.

Die ersten Systeme, die in sehr bescheidenem Umfang natürliche Sprache als Eingabesprache für ein Computersystem zuließen und aus diesen natürlichsprachlichen Formulierungen automatisch Informationen erschlossen, entstanden mit BASEBALL (GREEN et al., 1961), STUDENT (BOBROW, 1964), SIR (RAPHAEL, 1964) und ELIZA (WEIZENBAUM, 1966).

Teilaspekte der automatischen Sprachverarbeitung wurden durch die Entwicklung von Parsing-Systemen zur syntaktischen Analyse (KAY-Parser, 1967, Transition-network-Parser von WOODS, 1970) und von semantischen Netzwerken (QUILLIAN, 1968) vorangetrieben. Integrierte Systeme zur Informationsverarbeitung und Problemlösung, die komplexere Eingabeformulierungen in natürlicher Sprache zuließen und diese mit verfeinerten Mitteln analysierten, entstanden

den ab Beginn der siebziger Jahre in Form von Frage-Antwort-Systemen (SIMMONS, 1973, LUNAR-System von WOODS et al., 1972) und von durch natürlichsprachliche Eingaben manipulierbaren Mikrowelt-Systemen (WINOGRAD, 1972). Parallel zur Analyse von Fragmenten der geschriebenen Sprache liefen die Projekte auf dem Gebiet "Speech Analysis" (NEWELL et al., 1971, WOODS et al., 1976). Die Erfahrungen, Techniken und Programmierinstrumentarien, die aus der Forschung der sechziger Jahre resultierten, ermöglichten, intelligentes Verhalten nicht nur unter mathematisch-logischem oder technologischem Aspekt zu betrachten, sondern in den originären Kontext kognitiver Psychologie zu stellen. Dies führte nicht nur zu einer Reihe psychologisch orientierter Arbeiten (vgl. SCHANK/COLBY, 1973), sondern auch zu Versuchen interdisziplinärer Problemlösungstheorien wie etwa von NEWELL/SIMON, 1972.

Besondere Beachtung fand zwischen 1960 und 1970 die Roboterkonstruktion, das für den Laien faszinierendste Gebiet der KI. In die Roboterkonstruktion gehen die Erkenntnisse aus dem Gebiet "Machine Vision" und "Analyse gesprochener Sprache" ein, wenn Systeme entwickelt werden, die über optische und/oder akustische Perzeptoren zur visuellen Szenenanalyse und zur Spracherkennung verfügen. Die Betätigung mechanischer Greifer bzw. die Ortsveränderung mithilfe von funkgesteuerten Fahrwerken ermöglichen es dem Roboter, auf seine Wahrnehmungen mit Aktionen zu reagieren. Ein solcher Roboter ist somit augenscheinlicher Beweis für gelungene intellektuelle Leistungen der Perzeption, Kognition und der Handlungsumsetzung.²

1.2. Forschungsschwerpunkte

Die Forschungsschwerpunkte, um die sich die KI-Systeme und die ihnen zugrundeliegenden theoretischen Ansätze gruppieren, sind somit

- Computersimulation von Spielen
- Automatisches Theorembeweisen (mit Automatischer Programmierung)
- Roboterkonstruktion

- Computersimulation der visuellen Wahrnehmung, Mustererkennung und Szenenanalyse
- Analyse der natürlichen Sprache
- Informationsverarbeitung, Simulation kognitiver Prozesse.

Diese Forschungsbereiche sind jedoch nicht streng getrennt, vielmehr sind neben dem gemeinsamen Forschungsinteresse gleichartige Verfahren in der Behandlung von Daten über komplexe Wissensbereiche verbindendes Element für mehrere der Einzelschwerpunkte. Darüberhinaus sind häufig erst durch Integration von Einzeldisziplinen bestimmte KI-Projekte realisierbar: So führen z.B. Ergebnisse aus den Gebieten Denkspiele, Automatisches Theorembeweisen und Roboterkonstruktion zu dem Versuch, "common sense reasoning" nachzubilden, wie es etwa zum Erstellen eines einfachen Handlungsplans benötigt wird. Common sense reasoning ist seinerseits nach Meinung der KI-Forschung auch in der menschlichen Verständigung durch Frage und Antwort bzw. in der Produktion oder dem Verständnis eines Textes verborgen, insofern als Schlüsse gezogen werden, Ambiguitäten aufgrund von Plausibilitätsargumenten aufgelöst werden usw. Gemeinsam ist der KI-Forschung außerdem, daß ihre grundsätzliche Intention dahingeht, nicht nur theoretische Konstrukte zu entwerfen, sondern intelligente Systeme als Computerprogramme zu realisieren, die mit ihrer Umwelt, insbesondere auch den Menschen, die sie entwerfen, benutzen und verändern, durch Input- und Outputkanäle interagieren. Die Chance, in ein anwendungsorientiertes System überzugehen, das in anderen Gebieten wie etwa der Medizin, Chemie oder Verwaltung direkt eingesetzt wird, ist daher für KI-Entwicklungen gegeben, sofern sie eine gewisse Ausgereiftheit erreicht haben, die ihren Einsatz auch unter ökonomischen Gesichtspunkten rechtfertigt.

1.3. Zentren der KI-Forschung

Die ersten Aktivitäten im Bereich KI konzentrierten sich auf die Universitäten der US-amerikanischen Ost- und Westküste, wie das Massachusetts Institute of Technology (MIT) und die Stanford University, an denen z.T. speziell KI-Laboratorien eingerichtet wur-

den. Beteiligt an Forschungsunternehmen der KI sind aber auch nicht-universitäre Forschungsinstitute wie das Stanford Research Institute (SRI), das Xerox Palo Alto Research Center u.a.

Experimente mit Robotersystemen wurden u.a. am MIT (Greifer-System) und am SRI (mobiles System ohne "Hand") durchgeführt. Die Konstruktion des Roboters "Freddy" war Ergebnis der ersten intensiven Beschäftigung mit der KI in Europa, an der Universität von Edinburgh. Hier, ebenso wie an anderen britischen Universitäten, lag ein weiterer Schwerpunkt auf dem Gebiet der automatischen Beweisführung und der heuristischen Suche.

Die KI-Forschung auf dem europäischen Kontinent, die zunächst eher vereinzelt aufkam und deren Forschungsgruppen keine nennenswerten Kontakte untereinander pflegten, wandte sich u.a. in Frankreich, Italien, Schweden und der Bundesrepublik Deutschland neben der Verarbeitung natürlicher Sprache und der eher am Rand der KI angesiedelten Mustererkennung ebenfalls verstärkt dem automatischen Beweisen zu.³

2. Problemlösung als gemeinsamer Forschungsinhalt

2.1. Zum Problemlösungsbegriff

Die Bestimmungsversuche dessen, was intelligentes Verhalten ausmacht, sind kontrovers. Dennoch erschien es möglich, die Merkmale eines intelligenten Systems begrifflich näher zu bestimmen und damit die Fähigkeiten zu definieren, die KI-Systeme vor allen anderen Computerprogrammen auszeichnen. Ein solches Merkmal, das natürliche und künstliche Intelligenz verbindet, wurde sehr früh schon in der Fähigkeit zur P r o b l e m l ö s u n g gesehen.

Zur näheren Bestimmung dieses Terminus kann man ausgehen von einem Aufgabenbereich und einer Menge von Aktionen, die in diesem Aufgabenbereich möglich sind. Eine P r o b l e m s t e l l u n g beschreibt dann eine Ausgangssituation und eine Zielsituation, wobei die P r o b l e m l ö s u n g darin besteht, daß durch

geeignete Auswahl und Verkettung von Einzelaktionen von der Ausgangssituation aus die Zielsituation erreicht wird.

Ein typisches und in der Literatur immer wiederkehrendes Beispiel zur Illustration dieses Problemlösungsbegriffs ist das Problem von den "Missionaren und Kannibalen" (vgl. AMAREL, 1968):

Drei Missionare und drei Kannibalen wollen einen Fluß überqueren. Sie haben ein Boot, das höchstens zwei Personen aufnehmen kann, wobei sowohl die Missionare als auch die Kannibalen das Boot steuern können. Sobald auf einem der Ufer oder im Boot die Zahl der Missionare niedriger ist als die der Kannibalen, fallen letztere in ihre frühere Gewohnheit zurück und bemächtigen sich der Missionare.

Gesucht ist der einfachste Transportplan, nach dem alle Personen sicher auf das andere Ufer kommen.

An diesem Beispiel lassen sich die beiden zentralen Problemstellungen der KI-Forschung aufzeigen: das Problem der **R e p r ä s e n t a t i o n** und das der **Z i e l e r r e i c h u n g s s t r a t e g i e**.

2.2. Das Repräsentationsproblem

Bevor die Problemlösung algorithmisch nachgebildet werden kann, muß zuerst die Problemsituation und damit der Weltausschnitt, den Problemsituation und Lösungssituation tangieren, dargestellt werden. Eine für den Problembereich hinreichend adäquate Repräsentation muß einmal ermöglichen, Situationen zu beschreiben, etwa durch Darstellung der Objekte und der Beziehungen, die zwischen den Objekten bestehen. Zum anderen müssen Aktionen beschreibbar sein, die Situationen verändern, und das Ergebnis dieser Veränderungen. Eine solche Beschreibung von Situationen und Gesetzmäßigkeiten kann man dann als einen Teil des "Wissens" des intelligenten Systems betrachten, und es ist einleuchtend, daß die Darstellung von Wissen zu den grundlegenden Fragestellungen der KI-Forschung gehört.

Beim Entwurf einer Repräsentationsform für die Problemstellung des Beispiels der "Missionare und Kannibalen" sind u.a. folgende Faktoren zu berücksichtigen: Es müssen zwei Objektklassen unterschieden werden, von denen nur eine die Kannibalismus-Eigenschaft hat. Die Anzahl der Elemente beider Klassen muß in die Repräsentation eingehen, ebenso wie die Kapazität des Transportmittels. Die Kannibalismus-Bedingung, auffaßbar als das verbotene Zahlenverhältnis zwischen den Objektklassen, das durch keine Aktion erreicht werden darf, ist die grundlegende Gesetzmäßigkeit der Situation. Die einzige vollziehbare Aktion ist der Transport. In der Repräsentation zu vernachlässigende Eigenschaften des Beispiels sind u.a. die Benennungen der Objektklassen und die konkreten Situationsbedingungen, daß es sich z.B. um einen Fluß handelt oder daß ein Boot als Transportmittel dient.

2.3. Lösungsstrategien

Das zweite zentrale Problem bei der Konstruktion intelligenter Systeme ist die Auffindung und Formulierung von Zielstrategien oder Aktionsplänen.

In unserer Beispielpproblemstellung läßt sich die Zielerreichungsstrategie aus der genannten Gesetzmäßigkeit und der Kapazität des Transportmittels ableiten. Sie besteht darin, jeweils durch Transport und Rücktransport einer möglichst großen Anzahl von Kannibalen oder Missionaren nur solche Situationen eintreten zu lassen, in denen die Kannibalismus-Bedingung nicht erfüllt ist.

Als generelles Verfahren für die Ermittlung von Zielstrategien sei hier eines von mehreren bekannten erwähnt. Denkt man sich die Ausgangssituation einer Problemstellung als Wurzelknoten eines Baumes, die Folgesituationen, die durch einen Aktionsschritt aus der Ausgangssituation erzeugt werden können, als Nachfolgerknoten des Wurzelknotens, dann kann dieses Verfahren so lange fortgesetzt werden, bis ein Knoten des so entstehenden Lösungsbaumes einer Zielsituation der Problemstellung entspricht. Erfordert der Weg von Ausgangssituation zu Zielsituation eine größere Anzahl von Aktionsschritten und gibt es insgesamt eine nicht geringe

Zahl möglicher Aktionen, so entsteht ein Lösungsbaum von beträchtlicher Größe.

Soll ein intelligentes System bei begrenzter Speicher- und Zeitkapazität eine Problemlösung erreichen, können solche Lösungsbäume nicht systematisch aufgebaut und durchsucht werden. Einen wichtigen Platz nehmen deshalb in der KI-Forschung die Arbeiten an **H e u r i s t i k e n** ein, mit deren Hilfe möglichst nur solche Zweige eines Lösungsbaumes aufgebaut oder durchsucht werden, die mit großer Wahrscheinlichkeit zu einer Lösung führen.

Etikettiert man die Knoten des Baumes mit Situationsbeschreibungen und die Kanten mit Aktionsnamen, so kann man den Weg durch den Lösungsbaum, der vom Wurzelknoten bis zum Knoten einer Zielsituation führt, als **P l a n** für die Problemlösung betrachten. Dieser Plan kann im systemexternen Aufgabenbereich realisiert werden (etwa von den Missionaren im o.a. Beispiel!). Führt das intelligente System die Pläne selbst aus - als Roboter oder als spielender Automat -, kann man es als handelndes System auffassen; gibt jedoch das System den Lösungsplan nur als Ratschlag an den Systembenutzer, soll es als handlungsantizipierend klassifiziert werden.

Mit dem angedeuteten Problemlösungsbegriff lassen sich intelligente Systeme als solche Systeme kennzeichnen, die über ein Modell (als möglicherweise vereinfachte symbolische Darstellung und nicht im modelltheoretischen Sinn) ihrer Umwelt oder ihres Aufgabenbereichs verfügen und in der Lage sind, im Rahmen dieses Modells Pläne zu entwickeln (vgl. MICHIE, 1971).

Systemintern auf der Ebene der Repräsentation betrachtet, kann eine Situationsbeschreibung ein Satz oder eine Folge von Sätzen in einer formalen Repräsentationssprache sein. Den möglichen Aktionen zur Situationsveränderung oder den Gesetzmäßigkeiten entsprechen ebenfalls Sätze und/oder Prozeduren zur Veränderung von Sätzen oder zur Erzeugung neuer Sätze.

Unter diesem Gesichtspunkt präsentiert sich Problemlösung als die Aufgabe, im Rahmen eines formalen Systems diejenigen Ableitungsschritte zu finden, über die ein bestimmter Satz ($\hat{=}$ Zielsituation) aus einer gegebenen Menge von Sätzen ($\hat{=}$ Ausgangssituation) abgeleitet werden kann. Die besondere Rolle von Prädikatenkalkülen und

Theorembeweisern in der KI-Forschung ist damit angedeutet.

Interpretiert man die Sätze, die bei der beschriebenen Art der Problemlösung im Rahmen von Prädikatenkalkülen oder anderen formalen Systemen abgeleitet werden, nicht als Handlungsanweisungen, sondern als Propositionen über Sachverhalte, so kann man diese Problemlösung im engeren Sinn als Informationerschließung auffassen. Informationerschließende Problemlöser wären demnach zumindest nicht in der Lage, Handlungen auszuführen.

3. Sprachorientierte KI-Forschung

3.1. Problemlösung im Bereich sprachorientierter KI-Forschung

Der Zusammenhang zwischen künstlicher Intelligenz und natürlicher Sprache ergibt sich zum einen unter dem Gesichtspunkt der Verwendung natürlicher Sprache als einem naheliegenden Formulierungsmittel für Aussagen über irgendwelche Wirklichkeitsbereiche und zur Beschreibung von Problemstellungen in diesen Bereichen. Zum anderen kann man die Verständigung durch sprachliche Äußerungen sicherlich zu den Fertigkeiten rechnen, die Intelligenz erfordern.

Was ist nun gemeint, wenn in der KI-Literatur von "sprachverstehenden" Maschinen geredet wird?

Ein Automat "versteht" insofern natürliche Sprache, als er in der Lage ist, den Inhalt deklarativer natürlichsprachlicher Formulierungen über einen Weltausschnitt in Form seiner internen formalen Repräsentation zu reformulieren und so eine symbolische Darstellung des Weltausschnitts aufzubauen. Aufgrund der "Kenntnis" von Gesetzmäßigkeiten des Weltausschnitts besitzt der Automat Inferenzfähigkeit und kann natürlichsprachliche Fragen beantworten oder auf Aufforderungen sinnvoll reagieren. In vielen Fällen wird als Reaktion des Automaten eine natürlichsprachliche Antwort angestrebt.

Verstehen und Erzeugen natürlichsprachlicher Sätze sind erste Schritte zu einer "Simulation natürlichsprachlich kommunikativen

Verhaltens". In dieser Aufgabenstellung kann die generelle Thematik sprachorientierter KI-Forschung gesehen werden. Soll kommunikatives Verhalten simuliert werden, so ist allerdings das faktenbezogene Verständnis von natürlichsprachlichen Äußerungen im Sinne des Aufbaus und Abfragens einer Modellwelt dahingehend zu erweitern, daß Sprecherintentionen und Hörererwartungen, Kontextzusammenhänge des inner- und außersprachlichen Kontextes, Sprechaktmarkierungen und Argumentationsstrukturen in Texten und Dialogen verstanden und reproduziert werden. Dieses hochgesteckte Ziel, von dem der gegenwärtige Entwicklungsstand noch weit entfernt ist, wird wohl nur exemplarisch und für bestimmte vorab eingegrenzte Typen der Kommunikation, für die die linguistisch-pragmatische Forschung Evidenz erbracht hat, erreicht werden können.

Durch die Zielsetzung, Interaktion zu simulieren, kommt gleichzeitig der Problemlösung ein neuer Stellenwert zu, denn sprachliche Interaktion ist an sich ein problemlösendes Unterfangen: Dialoge werden geführt, u.a. um Informationen zu gewinnen, Streitfragen zu klären, Strategien für praktisches Handeln im Gespräch vorab zu erarbeiten. Die linguistische Analyse wird aus ihrer Isolation gelöst und in die Analyse der aufgrund der vollzogenen Sprachhandlungen möglichen Interaktionen der Dialogpartner einbezogen. Gleichzeitig wird der Bezug sprachlicher Interaktion zum Geschehen in der Außen- bzw. Modellwelt hergestellt.

3.2. Grundsätzliche Probleme

Der Konstruktion sprachverarbeitender Systeme liegt die Annahme zugrunde, daß natürliche Sprachen aufgrund ihrer Mehrdeutigkeiten und ihrer Paraphrasenmöglichkeiten zur systeminternen Wissensformulierung wenig geeignet sind. Es wird deshalb nach formalen Darstellungsmitteln gesucht, in denen sich der Inhalt natürlichsprachlicher Äußerungen reformulieren läßt und über denen formale Deduktionsmechanismen operieren können.

Aus den Anforderungen, daß das idealtypische sprachverarbeitende System in der Lage sein soll, natürliche Sprache zu analysieren und zu verstehen, die formulierten Problemstellungen zu lösen und

natürlichsprachlich die gefundenen Lösungen an den Dialogpartner zurückzugeben, folgen somit die vier **S c h w e r p u n k t e** der sprachorientierten KI-Forschung:

- (1) das Problem der Repräsentation des **I n h a l t s** natürlichsprachlicher Formulierungen.
- (2) das Problem der Formulierung von **I n f e r e n z - r e g e l n** und Inferenzmechanismen.
- (3) die linguistische **A n a l y s e** natürlichsprachlicher Formulierungen und ihre Operationalisierung mit dem Ziel der Reformulierung in der internen Repräsentationsform.
- (4) die **E r z e u g u n g** natürlichsprachlicher Formulierungen aus der internen Repräsentationssprache beim Dialog mit dem Systembenutzer.

Es erweist sich, daß die sprachorientierte Richtung zumindest bezüglich Schwerpunkt (1) und (2) sehr vieles gemeinsam hat mit nicht-natürlichsprachlich orientierten Zweigen der KI-Forschung. Es ist darüberhinaus durchaus möglich, auch in (3) und (4) typische Problemlösungsaufgaben zu sehen, insofern als die jeweils ausgangssprachlichen Formulierungen als Ausgangssituationen aufgefaßt werden können, denen aufgrund wohldefinierter Analyse- bzw. Syntheseaktionen Zielsituationen, d.h. die zielsprachlichen Formulierungen, zugeordnet werden.

3.2.1. Repräsentation des Inhalts natürlichsprachlicher Formulierungen

Da das Schema für die Repräsentation von Wissen nicht nur Basis für den Inferenzformalismus ist, sondern auch Zielpunkt der Analyse natürlichsprachlicher Formulierungen und Ausgangspunkt für ihre Generierung, kommt dem Repräsentationsproblem in sprachorientierten Systemen eine zentrale Bedeutung zu.

Natürlichsprachliche Formulierung und Entsprechung in der internen Repräsentationssprache müssen die gleichen Inhalte, zumindest bezüglich derjenigen Inhalte, mit denen das System befaßt ist, wiedergeben. Auf diese Bedingung, die durch die Übersetzungsrela-

tion erfaßt werden soll, wird in den Einzelbeiträgen von WULZ bzw. ZIFONUN in diesem Band näher eingegangen.

Drei Hauptansätze für Repräsentationsschemata scheinen sich in der Forschung herauszukristallisieren, die man verkürzt den logischen Ansatz, den Netzwerkansatz und den prozeduralen Ansatz nennen kann.

Dem l o g i s c h e n A n s a t z liegt die Verwendung prädikatenlogischer und -kategorialer Sprachen (vgl. MONTAGUE, 1972, 1973 und CRESSWELL, 1973) zur Darstellung von Wissen über eine Welt zugrunde, wobei ein Hauptproblem darin liegt, durch Erweiterung der Syntax und Semantik herkömmlicher Logiksprachen und durch geeignete Definition des Vokabulars die Ausdrucksmöglichkeiten nachzuahmen. Kennzeichnend für diesen Ansatz sind jedoch weniger die verwendeten und entwickelten prädikatenlogischen Sprachen als vielmehr ihre Verwendung im Rahmen von Kalkülen und ihre modelltheoretische Interpretation.

Die Verfechter dieses logischen Ansatzes führen als Vorteil dieses Verfahrens die Wohldefiniertheit an und die Möglichkeit, auf die Erkenntnisse und Einsichten einer langen Forschungstradition aus dem Bereich der Logik aufbauen zu können.

Die in diesem Band vorgestellte Konstruktsprache KS (vgl. den Beitrag von ZIFONUN) ist als Repräsentationssprache dieses Typs zu betrachten, denn sie stellt eine Erweiterung der Prädikatenlogik erster Stufe dar.

Überspitzt formuliert unterscheidet sich der N e t z w e r k - a n s a t z vom logischen Ansatz mehr akzidentiell als substantiell. In der Tat wurde auch verschiedentlich der enge Zusammenhang der beiden Ansätze herausgestellt (vgl. SIMMONS/BRUCE, 1971, neuerdings auch SCHUBERT, 1976 mit einer informativen Darstellung einiger Probleme dieses Ansatzes; für eine allgemeine Darstellung vgl. WOODS, 1975).

Das Hauptmerkmal des Netzwerkansatzes ist die Darstellung von Repräsentationseinheiten, sogenannten "concepts" als Knoten in einem Graphen, wobei in den verschiedenen Vorschlägen unterschiedliche Definitionen (manchmal leider auch gar keine) für Konzepte angegeben werden. Üblicherweise können Konzeptknoten Individuen,

Mengen und Eigenschaften bezeichnen. Es liegt auf der Hand, daß die zweidimensionale Darstellung von Netzwerken komplexe Zusammenhänge leichter durchschaubar und für den Betrachter verstehbarer machen als etwa lineare Darstellungen in einer prädikatenlogischen Sprache. Es mag nicht zuletzt der illustrativen Kraft dieser Darstellungsmethode zuzuschreiben sein, daß sie eine weite Verbreitung gefunden hat.

Eine Besonderheit des `prozeduralen Ansatzes`, der im wesentlichen auf WINOGRAD und HEWITT (vgl. WINOGRAD, 1972 und die dort zitierte Literatur) zurückzuführen ist, ist die Einbettung der Repräsentation in Programme. Während Objekte, Eigenschaften und Relationen ähnlich notiert werden wie in einer prädikatenlogischen Sprache, werden die in einer Welt möglichen Aktionen von den Relationen unterschieden und als programmierte Prozeduren dargestellt.

Das gleiche Verfahren wird auch zur Deduktion verwendet, was dazu führt, daß die Generalität der Deduktionsmechanismen, die im Rahmen von logischen Kalkülen anzutreffen ist, weitgehend zugunsten von Heuristiken aufgegeben wird.

Der große Vorteil dieses Ansatzes liegt in der außerordentlich ökonomischen Inferenzleistung. Dem steht jedoch der Nachteil einer für den Nichtspezialisten weniger leicht verständlichen Darstellung gegenüber und die Schwierigkeit, bei größeren Systemen noch die vielfältigen Interaktionen der Prozeduren zu überblicken.

Die drei Ansätze zur Darstellung des Wissens werden im Beitrag von ZIFONUN vor allem unter dem Gesichtspunkt der Interpretation der verwendeten Formalismen verglichen.

Nicht in unmittelbarer Abhängigkeit von einem der drei Repräsentationsansätze, immerhin doch in größerer Nähe zur prozeduralen Beschreibung befindet sich die sogenannte "`frame`"-Konzeption.

Mit ihr wird versucht, sich dem Problem der Strukturierung und internen Organisation des "Weltwissens" zumindest begrifflich zu nähern. Es wurde festgestellt, daß gewisse Regularitäten und Einzel-tatsachen nur innerhalb bestimmter Rahmenbedingungen Gültigkeit haben. Solche Rahmenbedingungen konstituieren die festen Umrisse va-

riabler Situationen bzw. die stereotypen Grundmuster wechselnder Einzelszenen. In einem zentralen Aufsatz hat MINSKY, 1975 versucht, eine allgemeine Theorie solcher Rahmenbedingungen ("frames") für die kognitive Organisation des Wissens und die Repräsentation des Wissens in KI-Systemen zu formulieren, deren wesentliche Merkmale er so beschreibt:

Here is the essence of the frame theory: When one encounters a new situation (or makes a substantial change in one's view of a problem), one selects from memory a structure called a frame. This is a remembered framework to be adapted to fit reality by changing details as necessary.

A frame is a data-structure for representing a stereotyped situation like being in a certain kind of living room or going to a child's birthday party. Attached to each frame are several kinds of information. Some of this information is about how to use the frame. Some is about what one can expect to happen next. Some is about what to do if these expectations are not confirmed.

We can think of a frame as a network of nodes and relations. The "top levels" of a frame are fixed, and represent things that are always true about the supposed situation. The lower levels have many terminals -- "slots" that must be filled by specific instances or data. Each terminal can specify conditions its assignments must meet. (The assignments themselves are usually smaller "sub-frames".) Simple conditions are specified by markers that might require a terminal assignment to be a person, an object of sufficient value, or a pointer to a sub-frame of a certain type. More complex conditions can specify relations among the things assigned to several terminals.⁴

Die Bedeutung von frame-ähnlichen Formalismen für das Verstehen von Texten wurde neben anderen von CHARNIK, 1975 und SCHANK, 1975b demonstriert, wobei CHARNIK schon in früheren Arbeiten insbesondere die Möglichkeiten für die Disambiguierung nominaler und pronominaler Referenzen untersucht hat.

Die Berücksichtigung übergreifender semantischer Strukturen der Situationskonstitution hat in der Linguistik selbst noch kaum Eingang gefunden. Hinweise finden sich in FILLMORE, 1977. Sein Begriff der "Szene", entwickelt auf der Basis der Kasustheorie, hat starke Anklänge an MINSKY's frame-Definition:

The study of semantics is the study of the cognitive scenes that are created or activated by utterances. Whenever a speaker uses a n y of the verbs related to the commercial event, for example, the entire scene of the commercial event is brought into play - is 'activated' - but the particular word chosen imposes on this scene a particular perspective.⁵

3.2.2. Inferenzregeln und Inferenzmechanismen

Soll ein System in die Lage versetzt werden, Schlußfolgerungen aus gespeicherten Informationen zu ziehen, so liegt der klassische Ansatz hierzu in der Anwendung logischer Deduktionsverfahren wie des oben erwähnten Resolution-Verfahrens. Logische Ableitung setzt voraus, daß im System neben Einzeldaten auch Gesetzmäßigkeiten, die in der dargestellten Welt gelten, in Form von nicht-logischen Axiomen in der internen Repräsentationssprache repräsentiert sind. Solche nicht-logischen Axiome können sich auf spezifische Regularitäten einer Mikrowelt beziehen, wie etwa in WINOGRAD's System über Spielzeugblöcke die Tatsache, daß alle Blöcke rot, grün oder blau sind. Sie können sich aber auch auf unseren Sprachgebrauch beziehen, der beim Prozeß des Spracherzeugens und -verstehens aktiviert wird. Ein Beispiel für eine solche Gesetzmäßigkeit ist unser Wissen, daß, wenn eine Person A einer Person B zu einem Zeitpunkt t etwas verkauft, die Person B ab Zeitpunkt t im Besitz dieses Objektes ist. Die sprachorientierte KI-Forschung hat sich vor allem auch mit dem Schlüsse-Ziehen ("inference making") aufgrund von Gesetzmäßigkeiten letzterer Art befaßt.

Problemlösung mittels *u n i f o r m e r* Deduktionsverfahren erwies sich bald als ineffizient, da es schon bei geringen Datenmengen aufgrund des rein syntaktischen Verfahrens zu einer kombinatorischen Explosion der im Deduktionsverfahren zu bearbeitenden Ausdrücke kommt.

Diese Ineffizienz konnte auch durch Verfeinerungen des Resolution-Prinzips, wie sie in CHANG/LEE, 1973 dargestellt sind, nicht grundsätzlich behoben werden. Deshalb wurden nun einerseits im Rahmen des logischen Kalküls heuristische Methoden eingesetzt, um die Suche nach geeigneten Ausdrücken zu verkürzen bzw. zu beschleunigen, und andererseits wurde mit dem prozeduralen Ansatz der Versuch präsentiert, durch die Einbettung von "Weltwissen" in prozedural zu verstehende Inferenzregeln den Ableitungsvorgang heuristisch zu steuern. Auch "frames" stellen in gewisser Hinsicht wissensspezifische Heuristiken dar.

Ein Theorembeweiser auf der Basis des Resolution-Prinzips wird in diesem Band im Beitrag von DILGER beschrieben.

3.2.3. Parsing und Überführung in eine interne Repräsentation

Bei der Überführung natürlichsprachlicher Äußerungen in Einheiten der internen Repräsentationssprache sind zwei Schritte als systematisch getrennt anzunehmen: die Analyse der ausgangssprachlichen Äußerungen und die Synthese der zielsprachlichen Äußerungen, d.h. der Formulierungen der Repräsentationssprache.

Die Analyse besteht in der Erzeugung einer syntaktischen (und semantischen) Strukturbeschreibung der natürlichsprachlichen Eingabe, aufgrund deren Übersetzungen in der Zielsprache synthetisiert werden, die den grammatisch-semantischen Bedingungen dieser - im Falle eines KI-Systems - formalen Kunstsprache genügen.

Das In-Beziehung-Setzen zweier Sprachsysteme im Überführungsprozeß ist ein theoretisch wie praktisch in der KI-Forschung nur ungenügend bearbeitetes Gebiet. Zwar liegen zum Parsing (im Sinne von syntaktischer Analyse) hinreichend ausgearbeitete Verfahren vor, die im wesentlichen in RUSTIN, 1973 dokumentiert sind; Verfahren zur eigentlichen Überführung sind jedoch erst im Unklaren umrissen (vgl. etwa RIESBECK, 1975 und WINOGRAD, 1972). Dies mag auch daher rühren, daß in der sprachorientierten KI-Forschung z. T. Ausdrücke der internen Repräsentation nicht als Ausdrücke einer anderen Sprache aufgefaßt werden, sondern als eine Art Tiefenstrukturen der entsprechenden natürlichen Sprache, die auch mehr oder weniger direkt das Repräsentationsvokabular in Form von semantischen Primitiven liefert. Bei dieser Auffassung hat die gesamte Überführung Parsing-Charakter, denn sie besteht in der Zuordnung einer semantischen Strukturbeschreibung, deren wesentliche Funktion darin liegt, semantische Ambiguitäten aufzulösen.⁶

Bei einer klaren Trennung von Parsing und Überführung wird dagegen die interne Repräsentation nicht als metasprachliche Beschreibung der Eingabe begriffen, sondern als Übersetzung in eine andere Objektsprache. Begreift man diese zweite Sprache als Semantik-

sprache für die natürliche Sprache, so bleibt ungeklärt, inwieweit eine ausgefaltete semantische Beschreibung der ausgangssprachlichen Äußerungen erforderlich ist, wenn die Erzeugung der internen Repräsentation, also der zweite Schritt der Überführung, als semantisches Interpretationsverfahren verstanden wird, in dessen Verlauf die voranalysierten Strukturen einander - im Medium der Übersetzungsentsprechungen - semantisch zugeordnet werden.

Im vorliegenden Band werden Parsing und Überführung in getrennten Beiträgen behandelt (vgl. die Beiträge von LÖTSCHER, LÖTSCHER/KOLVENBACH bzw. WULZ).

3.2.4. Generierung natürlichsprachlicher Formulierungen

Die Erzeugung natürlichsprachlicher Formulierungen aus Ausdrücken der internen Repräsentationssprache kann abstrakt als Überführung betrachtet werden, bei der die beiden Sprachen ihre Rollen als Ausgangs- und Zielsprache vertauscht haben. Sicher ist jedoch für das Verfahren bedeutsam, daß nun die morphologisch wesentlich ärmere und ambiguitätsfreie interne Repräsentationssprache analysiert und daß natürlichsprachliche Ausdrücke erzeugt werden sollen. Im Syntheseschritt wird sich daher an die Erzeugung satzkonstituierender inhaltstragender Elemente durch Übersetzung aus der internen Repräsentationssprache eine Komponente anschließen, die aus diesen Strukturen morphosyntaktisch wohlgeformte Äußerungen der natürlichen Sprache macht. Ob allerdings die im Analysealgorithmus verwendete Grammatik für die natürliche Sprache unverändert für den Synthesealgorithmus übernommen werden kann, ist in der Praxis zu zeigen. Bisher liegen auch zu dieser Überführungsaufgabe nur rudimentäre Ergebnisse der KI-Forschung vor, etwa die Ausführungen in SIMMONS, 1973 zur Synthese englischer Sätze aus semantischen Netzwerkrepräsentationen oder auch zur Erzeugung französischer Formulierungen aus einer Zwischensprache im Rahmen eines englisch-französischen maschinellen Übersetzungsprojekts in WILKS, 1973.

Der Aspekt der Sprachsynthese wird in den Beiträgen des vorliegenden Bandes nicht behandelt.

4. Sprachorientierte KI-Forschung und Linguistik

4.1. Abgrenzung der sprachorientierten KI-Forschung von Anwendungen der EDV in der Linguistik

Die Verwendung der Datenverarbeitung im Zusammenhang mit der Erforschung, Beschreibung oder dem instrumentalten Gebrauch einer natürlichen Sprache hat zu einer Fülle von Aktivitäten geführt, die u.a. unter den Namen Computerlinguistik oder Linguistische Datenverarbeitung zusammengefaßt werden. Diesem Bereich werden so unterschiedliche Arbeitsfelder wie die maschinelle Auswertung von Texten nach statistischen Gesichtspunkten, die automatische Lemmatisierung und Syntaxanalyse, Konkordanz- und Lexikonerstellung mit dem Computer oder auch die automatische Übersetzung, Teilaspekte des computerunterstützten Unterrichts, des automatischen Texteditings und der automatischen Dokumentation zugeordnet. Diesen weit divergierenden Aktivitäten ist gemeinsam, daß sie hilfswissenschaftliche Funktionen erfüllen: Es handelt sich um Arbeiten mit dem Computer für die Linguistik oder um Arbeiten für die Datenverarbeitung mit der Linguistik. Arbeit für die Linguistik ist auch die maschinelle Sprachanalyse, bei der der Computer als Testinstrument für Erzeugungs- oder auch Analysegrammatiken auf der Grundlage z.B. generativer Transformationsgrammatiken benutzt wird. Forschungsergebnisse auf diesem Gebiet werden in EISENBERG, 1976 vorgestellt.

Die linguistisch orientierte KI-Forschung, die man im Anschluß an UNGEHEUER, 1971 und neuerdings BATORI, 1977 mit entsprechender inhaltlicher Neubestimmung auch als linguistische Datenverarbeitung bezeichnen mag, versteht sich dagegen eher als eigenständiges interdisziplinäres Forschungsgebiet oder mehr noch als interdisziplinäre Forschungsmethode. Auf der Basis linguistischer, kommunikationswissenschaftlicher und computerwissenschaftlicher Forschung sollen Computersysteme entwickelt werden, in deren Mittelpunkt der Dialog zwischen Mensch und Computer zum Zwecke der Problemlösung steht. Diese Konzeption geht zurück auf das UNGEHEUER-sche Basismodell der Mensch-Maschine-Kommunikation. Da der linguistisch-kommunikative Teil solcher Systeme so ausgebaut

sein muß, daß "Verstehen" von natürlichsprachlichen Problemformulierungen und Erzeugen von natürlichsprachlichen Lösungsformulierungen gewährleistet ist, ist eine explizite linguistische Analyse (und Synthese) auf allen Ebenen der Sprache, also auf morphologischer, syntaktischer, semantischer und pragmatischer Ebene, erforderlich. Auch dadurch unterscheidet sich die Computersimulation von Sprachverhalten von den genannten Formen der Computerlinguistik, bei denen meist bestimmte sprachliche Teilaspekte - wie Lexikon oder Syntax - isoliert behandelt werden. Der methodische Rahmen der Computersimulation macht es zudem notwendig, daß eine solche Beschreibung formal und operational interpretierbar zu erfolgen hat.

Interdisziplinarität, Untersuchung von Kommunikationsprozessen im Spiegel der Computersimulation, Integration aller sprachlichen Analyseebenen innerhalb eines Systems, diese Bestimmungsgrößen sollten ausreichen, um die Eigenständigkeit der KI-orientierten LDV-Forschung zu begründen. Nicht notwendig erscheint es, LDV-Forschung darüberhinaus mit dem Anspruch zu betreiben, daß in ihr die "prozeduralen Aspekte der Sprache" erforscht werden sollen. Diese Zielsetzung schreibt BATORI, 1977 der LDV zu und versteht unter den prozeduralen Aspekten den P r o z e ß des Sprachverstehens und des Spracherzeugens.

4.2. Der Beitrag der sprachorientierten KI-Forschung zur Erforschung der Sprache

Der in den Überlegungen BATORI's enthaltende Anspruch, daß die linguistische Datenverarbeitung Beiträge zur Erforschung der Sprache leistet, die von der Linguistik selbst nicht geleistet werden (nicht geleistet werden können), ist zu überprüfen. BATORI argumentiert, die Systemlinguistik beschreibe quasi statisch die Strukturen natürlicher Sprachen, sie unterlasse es jedoch, das regelhafte Zusammenspiel und das planvolle Ineinandergreifen von Realisationen der Systemteile bei Spracherkennungs- und Spracherzeugungsprozessen zu beschreiben. Die Analyse dieses "prozeduralen" Aspektes sei von qualitativ völlig anderer Art als die sy-

stemlinguistische Beschreibung und werde in der algorithmischen Sprachanalyse der LDV geleistet. Dem ist entgegenzuhalten, daß algorithmisches Vorgehen kein Merkmal der Sprachanalyse oder -synthese ist, sondern vor jedem Gegenstand durch die Bedingungen der Verarbeitung mit dem Computer gegeben ist. Denn algorithmisch vorgehen heißt, ein automatisches Ablaufverfahren zur Lösung einer Problemstellung angeben, das sich seinerseits als formales System im mathematischen Sinne präzisieren läßt. Da ein formales System aus einem Alphabet, einer Menge von über diesem Alphabet bildbaren Formeln, einer Menge von Axiomen und einer Menge von Regeln besteht, mit deren Hilfe gewisse Ausdrücke abgeleitet oder erzeugt werden können, wird deutlich, daß dieser Begriff des formalen Systems so allgemein ist, daß formale (generative) Grammatiken und Logikkalküle ebenso unter ihn fallen wie beliebige berechenbare Funktionen.

Daraus ergibt sich, daß es Gegenstände - auch im Bereich der Linguistik - gegeben hat und weiterhin gibt, die sich als formales System, und somit algorithmisch, beschreiben lassen, ohne daß damit eine Simulation eines mentalen Prozesses impliziert würde, ja daß gerade auch solche Teilbereiche wie die Grammatiktheorie, die BATORI zu den nicht-prozeduralen Aspekten der Sprache zählt, sich mit diesen Mitteln darstellen lassen. Die Vermutung, daß kein direkter Bezug zwischen Algorithmisierung und sprachlichem Prozeß besteht, erhärtet sich auch am Beispiel: So ist die von BATORI genannte Frage "Wie erkennt man das Ende einer Nominalphrase?" eher eine Frage nach der Verwendung von Struktureigenschaften von Nominalphrasen in einem Algorithmus als eine Frage nach dem Prozeß des Verstehens oder Erzeugens einer Nominalphrase.

DRESHER/HORNSTEIN, 1976 messen einige prominente Arbeiten der sprachorientierten KI genau an diesem Anspruch, einen originären Beitrag zur wissenschaftlichen Erforschung der Sprache, im besonderen der sprachlichen Performanz im Verstehensprozeß, zu leisten. Ihre detaillierte Kritik von WINOGRAD, 1972, MINSKY, 1975, SCHANK, 1972 und 1973b und des Augmented-Transition-Network-Ansatzes in WANNER/MARATSOS, 1974 bzw. in WANNER/KAPLAN/SHINER, 1975 hat überwiegend negative Ergebnisse. DRESHER/HORNSTEIN berufen sich auf den CHOMSKYSchen Kompetenzbegriff und messen die Qualität linguistischer Theorien an den Kriterien der "observational",

"descriptive" und "explanatory adequacy" von CHOMSKY, 1957.

Die Anforderungen, die sie an eine linguistische Theorie der Performanz stellen, sind letztlich am Kriterium der Erklärungsadäquatheit ausgerichtet. So soll der Gegenstandsbereich desjenigen Teils der Performanzforschung, der sich auf eine Theorie des "processing of language in real time" bezieht, der folgende sein:

the elaboration of certain general principles, which, in this case, underlie the functioning of the various components in a processor and the interactions among them.⁷

DRESHER/HORNSTEIN argumentieren nun, daß die genannten KI-Arbeiten kaum die Ebene der Beobachtungsadäquatheit in der Beschreibung von Sprachprozessen erreichen; keineswegs seien sie stringent in ihren Hypothesen über menschliches Sprachverstehen und Spracherzeugen.⁸

Trotz der vernichtenden Kritik gehen DRESHER/HORNSTEIN und bestimmte Vertreter der kritisierten KI-Forschung von einer gemeinsamen Grundhypothese aus, nämlich der Hypothese, daß Spracherklärung letztlich nur durch die Aufklärung "mentaler Realitäten oder kognitiver Prozesse" geleistet werden kann. Während DRESHER/HORNSTEIN u.E. zu Recht feststellen, daß eine solche Erklärung der Computersimulation bisher nicht gelungen ist - nicht gelingen kann, wie wir aufgrund der Analyse algorithmischen Vorgehens sahen - , ist die gemeinsame Grundhypothese ihrerseits fragwürdig. Weist man jedoch die Forderung zurück, daß linguistische Theorien angeborene mentale Fähigkeiten erklären sollten, kann die Einsicht, daß Computersimulation sprachlicher Interaktion keine Simulation kognitiver Prozesse bedingt, nicht als Einschränkung empfunden werden.

Wohl zu überprüfen ist jedoch, in welchem Verhältnis ein Simulationsmodell zu der Realität, die es simuliert, steht. Wie HARBORDT, 1974 ausführt, ist die Ähnlichkeit des Input-Output-Verhaltens von zu simulierender "Wirklichkeit" und Modell zwar eine Voraussetzung, nicht jedoch eine Garantie für die Strukturähnlichkeit zwischen beiden. Ein Simulationsmodell, das Strukturen des Gegenstandes "Sprachverhalten" abbildet, muß die zugrundeliegenden Regeln, nach denen sprachliches Verhalten abläuft, ebenso

nachvollziehen wie das auf Input und Output reduzierte sprachliche Verhalten selbst. Dieser Nachvollzug sprachlicher Regeln geschieht aufgrund gespeicherter Regelbeschreibungen, wie sie auch außerhalb der Computersimulation bei linguistischen Untersuchungen Anwendung finden. Solche Beschreibungen sind empirische Hypothesen über die Regeln, im Sinne von für eine soziale Gruppe gültigen Mustern, denen Sprecher und Hörer beim Gebrauch sprachlicher Einheiten in der Kommunikation folgen.¹⁰ Die mögliche konkrete Realisierung solcher Regeln in den Gehirnen der Kommunikationspartner ist hier ohne Interesse und hat kein Analogon in den Algorithmen, die Regelbeschreibungen in automatische Verfahren umsetzen.

Festzuhalten ist daher, daß es kein sprachliches Forschungsgebiet geben kann, das der LDV allein vorbehalten ist. Performanztheorien, sofern diese überhaupt sinnvoll erscheinen, oder Theorien der sprachlichen Kommunikation werden LDV-unabhängig entwickelt. Wichtig erscheint allerdings der methodische Aspekt, der die Computersimulation auszeichnet: Die Simulation in praxi hat nämlich eine andere Qualität als die rein theoretische Erklärung. Falsche Regelbeschreibungen führen anders als bei einer rein theoretischen Erklärung zu direkter Falsifikation in Form mißlungener Interaktionen, ebenso wie Lücken in der Beschreibung und Algorithmisierung von sprachlichen Regeln manifest und lokalisierbar werden.

Anders als bei der natürlichen Interaktion zwischen menschlichen Dialogpartnern sind darüberhinaus mißlungene Mensch-Maschine-Kommunikationen nicht möglicherweise Folge von akustischen oder anderen performanzbedingten Verständnisschwierigkeiten oder von Mißverständnissen, die auf unterschiedlichen Gebrauch sprachlicher Einheiten bei Sprecher und Hörer zurückgehen. Ein Erklärungsversuch für Dialogabläufe anhand belegter Beispiele hat daher solche Einflüsse zu berücksichtigen. Ein Computerdiallog ist dagegen das Modell einer "idealen" Kommunikation, in der Sprecher- und Hörerkompetenz aneinander angeglichen sind, insofern als ja der Sprecher sein sprachliches Verhalten an der explizit formulierten Kompetenz des Systems auszurichten hat. An solchen Interaktionen können also von den genannten Faktoren unbeeinflußt Dialogmuster durchgespielt werden.¹¹

Anmerkungen

- 1 Der erste offizielle Gebrauch des Begriffs "Künstliche Intelligenz" (KI) ("Artificial Intelligence") datiert in das Jahr 1956. Im Sommer 1956 wurde in Dartmouth, New Hampshire, eine Konferenz zum Thema KI abgehalten, deren Initiatoren M. Minsky, N. Rochester, C. Shannon und J. McCarthy waren; vgl. McCorduck, 1977.
- 2 Einen ausführlichen Überblick über Entwicklung und Schwerpunktsetzung in der KI, auf den sich die Darstellung hier stützt, gibt Nilsson, 1974.
- 3 Pitrat/Sandewall, 1977 beschreiben die Entwicklung der KI-Forschung in Westeuropa. Im besonderen wird auf den Beginn einer Koordination der Aktivitäten in der BRD durch die Gesellschaft für Informatik (GI) hingewiesen.
- 4 Minsky, 1975, 118.
- 5 Fillmore, 1977, 17.
- 6 Vgl. dazu die Ausführungen in Wilks, 1976a zu "Parsing English".
- 7 Drescher/Hornstein, 1976, 328.
- 8 Positiver ist die Einschätzung von Sgall, 1976: Wohl erkennt er die Schwächen von z.B. Winograd, 1972, was linguistische Theoriebildung und Generalisierbarkeit des Ansatzes angeht, dennoch hält er den Ansatz der sprachorientierten KI-Forschung für zukunftsweisend. Die Neuerungen, die von daher in die Linguistik eingebracht werden könnten, betreffen nach Sgall das Verhältnis zwischen Beschreibung der linguistischen Kompetenz und anderen Bereichen wie Performanztheorie, Pragmatik u.a.
- 9 Chomsky, 1969, 14.
- 10 Zu einer Explikation dieses Regelbegriffs vgl. Winch, 1966 und Heringer, 1975.
- 11 Mit theoretischen Gesichtspunkten der Simulation natürlich-sprachlicher Dialoge im Rahmen von KI-Systemen beschäftigt sich W. von Hahn in seinem Vortrag "Probleme der Simulationstheorie und Fragepragmatik bei der Simulation natürlich-sprachlicher Dialoge", gehalten auf der Arbeitstagung "Kognitive Psychologie" in Hamburg, April 1978. Im besonderen werden die Fragen der Erwartungen von Benutzern gegenüber der Dialogqualität im Dialog mit dem künstlichen Partner und der Bewertbarkeit von Simulationsmodellen als 'richtig' kritisch untersucht.

Der Vortrag konnte im vorliegenden Beitrag nicht mehr berücksichtigt werden. Im Anschluß an die Ausführungen von Hahns wäre festzuhalten, daß eine Anpassung des natürlichen Dialogpartners an die Kompetenz des Systems für einen echten Benutzer im anwendungsorientierten Fall problematisch ist. Dies trifft nicht zu für den Bearbeiter des Systems oder denjenigen, der ein Dialogsystem zu Forschungszwecken betreibt.

G.L. Berry-Rogghe

Hans Dieter Lutz

Kaija Saukko

DAS INFORMATIONSSYSTEM PLIDIS

1. Aufgabenstellung und Zielsetzung
 - 1.1. Allgemeine Darstellung
 - 1.2. Ziel eines problemlösenden Informationssystems mit Deutsch als Interaktionssprache
 - 1.3. Die wissenschaftliche Bedeutung von PLIDIS
 - 1.4. PLIDIS als Experimentalsystem
2. Der Einsatzbereich von PLIDIS
 - 2.1. Daten
 - 2.2. Problemstellungen
 - 2.3. Realisierungsstufen
3. Die interne Repräsentation für das Weltmodell 'Abwasserüberwachung'
4. Systemarchitektur
 - 4.1. Gesamtüberblick
 - 4.2. Die PLIDIS-Hauptkomponenten
5. Beispielanalyse einer Frage an PLIDIS
 - 5.1. Die morphologische Analyse
 - 5.2. Die syntaktische Analyse
 - 5.3. Die Überführung in KS
 - 5.4. Der Beantwortungsprozeß
6. Sprachumfang und Leistungsfähigkeit
7. Übertragbarkeit des Systems

Anmerkungen

1. Aufgabenstellung und Zielsetzung¹

1.1. Allgemeine Darstellung

Die Konzeption, nach der ein problemlösendes Informationssystem mit Deutsch als Interaktionssprache (PLIDIS) entwickelt werden kann, läßt sich durch die folgenden zugrundegelegten Annahmen charakterisieren:

- (1) Es ist möglich, eine formale Sprache zu konstruieren (Konstruktssprache KS), die zwei Funktionen erfüllt:
 - Semantikssprache zur formalen Darstellung des Inhalts von natürlichsprachlichen Äußerungen wie Wörter, Sätze, Texte zu sein,
 - Modellbeschreibungssprache zur Darstellung der Objekte, Beziehungen und Gesetzmäßigkeiten zu sein, die in einem Weltausschnitt (im Sinne von Sachbereich oder Problem-bereich) vorzufinden sind;
- (2) es ist möglich, Operationen über dieser Konstruktsprache zu definieren, die die Fähigkeit nachbilden, Sprache zu verstehen und Probleme zu lösen; Fähigkeiten dieser Art sind etwa: Verstehen als Auflösen von Mehrdeutigkeiten, Rückfragen, Folgerungen Ziehen, Antworten;
- (3) es ist möglich, Regeln einer Übersetzungsgrammatik anzugeben, die beschreiben, wie natürlichsprachliche Äußerungen in diese Konstruktsprache zu übersetzen sind und wie konstruktsprachliche Formeln in natürliche Sprache zurückzuübersetzen sind;
- (4) es ist möglich, im Rahmen eingeschränkter Lösungen für die eben genannten drei Annahmen sinnvolle Realisierungen auf einer Datenverarbeitungsanlage zu implementieren, d.h. Systeme, die nicht nur labormäßig Ausschnitte von Sprachverstehen und Problemlösung nachbilden, sondern auch in der Praxis erprobbar sind.

Aufgrund dieser Annahmen wurde ein PLIDIS konzipiert, das besteht aus

- einem linguistisch-logischen Teil, der die deutsch-sprachige Eingabe in eine Konstruktsprache überführt, in der die interne Repräsentation (IR) der Eingabe erfolgt, und
- einem Problemlösungsteil, der zum einen die üblichen Speicher- und Retrievalaufgaben übernimmt, zum anderen darüber hinaus die im Anwendungsbereich geltenden Gesetzmäßigkeiten in die Problemlösung miteinbezieht.

1.2. Ziel eines problemlösenden Informationssystems mit Deutsch als Interaktionssprache

PLIDIS orientiert sich in seiner Konzeption nicht primär an datentechnischen Zweckmäßigkeiten, sondern an den Aufgabenstellungen und Kommunikationsformen, die in einem Anwendungsbereich gegeben sind. Dies wird erreicht durch

- die Verwendung eines durch den jeweiligen Anwendungsbereich bestimmten fachsprachlichen Ausschnitt des Deutschen als Dialogsprache;
- die Fähigkeit, in einer Fachsprache formulierte Inhalte 'zu verstehen' und abzuspeichern;
- die Fähigkeit, dem Benutzer die Beibehaltung der eigenen Begriffsbildung zu ermöglichen, so daß dieser nicht gezwungen ist, seine Problemstellungen in eine durch die Funktionsweise eines Informationssystems bedingte Begriffsbildung (z.B. eine logisch-mengentheoretische Begriffsbildung mit Recordstrukturen als Prädikaten) zu übersetzen;
- die Fähigkeit, im Dialog mitgeteilte Problemstellungen gemäß den im Anwendungsbereich geltenden allgemeinen Problemsituationen zu erkennen und in Abfragen an die Menge der abgespeicherten Informationen umzuwandeln;
- die Fähigkeit, sowohl bei der Erkennung der Problemstellung als auch bei ihrer Lösung auf Informationen zurückzugreifen,

die nicht direkt mitgeteilt worden sind, sondern aus mitgeteilten Informationen ableitbar sind.

PLIDIS wird in einem Anwendungsbereich erprobt, der so definiert ist, daß er einen tatsächlich gegebenen Bedarf abdeckt. Grundsätzlich soll die Konstruktion des Systems so allgemein sein, daß die Übertragbarkeit in andere Anwendungsbereiche gegeben ist.

Die Vorgehensweise, auf die Aufgabenstellungen und Kommunikationsformen in einem Anwendungsbereich abzuheben und von daher ein System mit den oben beschriebenen Spezifika zu entwickeln, soll durch die folgende Überlegung plausibel gemacht werden.

Betrachtet man in der derzeitigen Praxis die Beziehung zwischen maschinell gespeicherten Daten und den Personen, die sich mit Gewinn solcher Datenbanksysteme zur Lösung ihrer Probleme bedienen könnten, so wird man feststellen, daß diese Beziehung durch eine Sprachbarriere gekennzeichnet ist. Diese Sprachbarriere beim Zugriff auf Datenbanken ist nur durch Spezialisten zu überwinden, die entweder als Datenverarbeiter ausgebildet sind oder doch zumindest im ständigen Training für den Umgang mit formalen Schemata stehen. Nur solche Schemata, die zudem von einem Datenbanksystem zum anderen stark variieren können, erlauben derzeit den Zugriff auf computergespeicherte Daten, wobei für einen Benutzer meist auch noch nähere Kenntnis der Datenstrukturen erforderlich ist.

Diese Situation hat in der Praxis zur Folge, daß das Potential von DV-Anlagen nicht unmittelbar in die Lösung von Problemen bei politischer, administrativer oder kaufmännischer Planung und Entscheidung einbezogen wird, sondern nur mittelbar durch DV-versierte Mittelsmänner zugänglich ist mit all den damit verbundenen Störfaktoren wie Zeitverzögerungen, Fehlinterpretationen u.ä.²

Diese Sprachbarrieren werden zumindest auf ein Minimum reduziert, wenn es dem Benutzer von Datenbanksystemen erlaubt ist, sich zur Benutzung eines derartigen Systems seiner Muttersprache in der ihm gewohnten fachsprachlichen Ausprägung zu bedienen.

1.3. Die wissenschaftliche Bedeutung von PLIDIS

Unter einem wissenschaftlichen Aspekt läßt sich die Konzeption und Entwicklung von PLIDIS in zweierlei Hinsicht charakterisieren. Zum einen wird mit PLIDIS der Versuch gemacht, Mängel konventioneller Informationssysteme wie experimenteller Informationssysteme (wobei letztere sich bislang im "Laborversuch-Stadium" befinden) zu überwinden. Experimentelle Informationssysteme erfüllen die an PLIDIS gestellten Anforderungen jeweils nur in Teilbereichen.³

Die Erfahrung hat gezeigt, daß Einschränkungen oder Erweiterungen in den meisten Fällen nicht nur die Konzeption und den Ausbau der direkt betroffenen Systemkomponenten berühren; derartige Einschränkungen oder Erweiterungen haben vielmehr sehr direkte Auswirkungen auf die Anforderungen an die Konzeption des Gesamtsystems. Daraus läßt sich ableiten, daß Systemmodifikationen schwerwiegender Art nur in den seltensten Fällen durch schrittweises Vorgehen in ein ursprüngliches System eingebracht werden können. Die Konsequenz daraus wiederum bedeutet, mit der Konzeption eines Systems ein möglichst breites Leistungsspektrum abzudecken und die für Teilbereiche vorliegenden Forschungsergebnisse daraufhin anzupassen bzw. aufbauend auf den vorhandenen Ergebnissen neue Lösungen zu erarbeiten.

Zum anderen liegt die Bedeutung des Unternehmens in der multidisziplinären Verknüpfung von Forschungsergebnissen aus den Bereichen der Linguistik, der Künstlichen Intelligenz und der Informatik.⁴

Die Versuche, Formen sprachlichen Handelns durch ein auf einem Rechner realisiertes System nachzubilden, sind motiviert durch die Annahme, daß die Umsetzung von Sprachverstehen und sprachlichem Reagieren in explizite Algorithmen tiefere Einsichten in die Struktur solcher dialogischer Abläufe und ihrer Einbettung in Handlungszusammenhänge ermöglicht.

Diese Annahme stützt sich darauf, daß Interaktionsprozesse, die im Falle einer gelungenen 'natürlichen' Interaktion "von selbst", d.h. aufgrund der Regelbeherrschung der Dialogpartner ablaufen,

im Falle einer Mensch-Maschine-Kommunikation auf formalisierten und explizit gespeicherten Regelbeschreibungen beruhen. Das Nachbilden von Interaktionsprozessen als 'künstliche' Mensch-Maschine-Kommunikation setzt daher die Analyse und exakte Beschreibung der Regeln des syntaktischen und semantisch-pragmatischen Gebrauchs aller sprachlichen Einheiten voraus, die in der Interaktion Verwendung finden. Falsche Regelbeschreibungen führen bei der Überprüfung auf dem Computer zu direkter Falsifizierung.

1.4. PLIDIS als Experimentalsystem

Die Einbeziehung pilothafter Anwendung, d.h. die Überprüfung gefundener Lösungen in der Praxis und die dazu notwendige Zusammenarbeit mit einem Anwender ermöglicht nicht nur eine Überprüfung der Tragfähigkeit erarbeiteter Lösungsansätze. Durch die Anbindung an eine konkrete Realität werden auch bereits Strukturierung und Darstellung eines Sachbereichs und die Erarbeitung von Interaktions- und Problemlösungsmöglichkeiten der Ausschließlichkeit individueller Vorstellungen entzogen, da die Ansprüche an ein System aus der Praxis, unabhängig von aprioristischen Theoriefragmenten abgeleitet werden müssen.

Diese Zusammenarbeit mit einem Pilotanwender kann jedoch nicht eine Legitimation für ad-choc-Lösungen zur Befriedigung individueller Ansprüche eines Anwenders darstellen. Auch bei der Ausarbeitung von Lösungsvorschlägen für spezielle Anwenderprobleme muß die Gesamtkonzeption und der wissenschaftliche Anspruch im Auge behalten werden. Nur so läßt sich die Übertragbarkeit einmal gefundener, auf größtmögliche Generalität hin konzipierter Lösungen auf neue, vom ursprünglich gewählten Bereich abweichende Anwendungsgebiete gewährleisten.

Neben diese Kennzeichnung des Experimentalcharakters von PLIDIS in Bezug auf die Generalisierung konkret in einem Anwendungsbe-
reich auftretender Problemfälle tritt ein anderes Merkmal: der modulare Aufbau des Systems. Die Modularität ist so ausgelegt, daß auf jeder Stufe der Problembearbeitung eine Interaktion mit

dem System möglich ist. Dies ist eine nicht zu unterschätzende Unterstützung bei der Lokalisierung von auftretenden Schwierigkeiten bei einzelnen PLIDIS-Komponenten. Darüber hinaus ist mit der Modularität die Voraussetzung geschaffen, das System und seine einzelnen Komponenten systematisch und effizient auszutesten.

2. Der Einsatzbereich von PLIDIS

Der Einsatzbereich, in dem PLIDIS zum gegenwärtigen Zeitpunkt entwickelt und erprobt wird, ist das Sachgebiet der Abwasserüberwachung der metallverarbeitenden Industrie. Voraussetzung für die Interaktion des Anwenders mit dem System PLIDIS zur Lösung seiner Probleme ist, daß der Anwender sein Sachgebiet, d.h. seine sach- und problembezogene Welt kennt. Dazu ist es erforderlich, daß das Wissen in diesem Sachgebiet so analysiert, strukturiert und beschrieben ist, daß der Benutzer des Systems dieses Wissen als empirischen Kontext in seinem Anwendungsbereich einsetzen kann. D. h.: Es müssen alle Informationen, die der Benutzer zum Erkennen und Lösen seiner praktischen Probleme benötigt, ermittelt und beschrieben sein. Die Informationen, die der Benutzer in dem begrenzten Anwendungsbereich der Abwasserüberwachung benötigt, bilden den Weltausschnitt dieses Anwendungsbereiches als Teil aller vorhandenen Informationen aus der gesamten Welt der Abwasserüberwachung bzw. des Umweltschutzes.

2.1. Daten

Die Daten eines Anwendungsbereiches müssen so strukturiert und beschrieben sein, daß alle notwendigen Informationen aus seiner Datenwelt abgefragt werden können. Die Datenstruktur im Bereich der Abwasserüberwachung ist durch seine Personen, Objekte, Sachverhalte und Funktionen vorgegeben. Die wichtigsten Daten in diesem Sachbereich bilden die Daten über Personen, Orte, Abwasserinhaltsstoffe und -eigenschaften und die Daten über die Normenkontrolle.

Im Einsatzbereich von PLIDIS sind P e r s o n a l d a t e n enthalten. Es handelt sich um Daten

- über Firmen und Betriebe, bei denen Abwasser anfällt,
- über Behörden, die die Normenkontrolle durchführen,
- über Labore, die die Abwasserproben prüfen und
- über Personen und Personenkörperschaften, die die Funktionen 'Probenehmer', 'Prüfer', 'Auswerter' haben.

Weiterhin sind die O r t s d a t e n zu beschreiben. Nach der Struktur der Welt der Abwasserüberwachung wird unterschieden nach

- 'Standort' (Firmenort, Behördenort, Laborort),
- 'Lageort' (Ort der Kläranlage, Ort der Mülldeponie usw.),
- 'Gebietskörperschaft' (Gemeinde, Kreis, Land, usw.) und
- 'Gewässer' (Vorfluter, Fluß, Strom, See, Meer).

Darüberhinaus werden die M e ß d a t e n für Abwasserinhaltsstoffe und -eigenschaften ermittelt: Es sind Daten über Meß- und Maßangaben, die für den Anwendungsbereich genau festgelegt sind.

Den zentralen Datenbereich im Anwendungsbereich der Abwasserüberwachung bilden die A b w a s s e r - u n d P r o b e w e l t - d a t e n . Diese Daten geben Informationen über

- Vorhandensein, Art und gegebenenfalls Menge bzw. Intensität bestimmter Abwasserinhaltsstoffe und -eigenschaften und über
- die Abwassernormen (Grenzwerte).

Die Informationen aus der Probewelt werden über den Probeentnahmebereich im Laborbericht als Abwasser- und Probewelt Daten festgehalten. Hierbei handelt es sich überwiegend um Daten über den Probenehmer, über Ort und Zeit der Probeentnahme, über die bei der Probeentnahme festgestellten Werte der Abwasserinhaltsstoffe und -eigenschaften, über Meßverfahren und -geräte, über Namen und Orte des Labors, über den Prüfer, über Prüfverfahren und -geräte und über Prüfergebnisse.

Je nach Einsatzbreite des Systems PLIDIS, die sich aus den Anforderungen des Anwenders ergibt, müssen die Datenbereiche erweitert werden wie z.B. um Daten aus den Bereichen Abwasserbilanz, Hilfstoffe, Abwasserrückverfolgung und Folgevorgänge.

2.2 Problemstellungen

Der Ausgangspunkt für die Problemstellungen in einem Anwendungsbereich sind die Aufgaben und Ziele des Benutzers in diesem Bereich.

Im Anwendungsbereich der Abwasserüberwachung ist eine vorrangige Problemstellung die Frage nach den Vorgängen in diesem Bereich. Es wird unterschieden zwischen Problemstellungen im Bereich

- des Abwasservorganges (produzieren; anfallen; reinigen; einleiten),
- des Überwachungsvorganges (probenehmen/messen; prüfen; auswerten/vergleichen),
- des Folgevorganges aufgrund der aus dem Überwachungsvorgang entstandenen Tatbestände: Einhaltung gesetzlicher Normen (neue Probe nehmen/messen usw.); Verstoß gegen gesetzliche Normen (Bescheide mit Auflagen und/oder Bußgelder erteilen; beseitigen; vorbeugen).

In seiner ersten Erprobungsstufe wird das System PLIDIS mit den Problemstellungen des Kernbereiches der Abwasserüberwachung, dem Überwachungsvorgang getestet. Hierbei werden an das System in erster Linie folgende Fragen gestellt:

Bei welcher Firma wurden wann, wie oft, von wem und nach welchem Verfahren Proben entnommen/welche Abwasserinhaltsstoffe und -eigenschaften gemessen?

Die Antworten auf diese Fragen ergeben sich aus den Vorgängen 'Probenehmen' und 'Messen':

Ein Probenehmer aus einer Überwachungsbehörde nimmt zu einem bestimmten Zeitpunkt eine Probe aus dem gereinigten Abwasser an der Probeentnahmestelle einer Firma am Ort der Firma durch ein Verfahren mithilfe bestimmter Geräte nach bestimmten Regeln und mißt durch ein Meßverfahren bestimmte Eigenschaften des Abwassers für den von ihm abzugebenden Probeentnahmebericht.

Die Angaben sind im Probeentnahmebericht festgehalten und weisen auf die eigentlichen Problemstellungen des Überwachungsvorganges hin, nämlich auf den Vergleich zwischen den Meßwerten und den Grenzwerten.

Welche Meßwerte für welche Stoffe und Eigenschaften und nach welchem Verfahren wurden wann von wem festgestellt?

Bei welchen Stoffen und Eigenschaften wurden die Grenzwerte wo und wann über- oder unterschritten?

Diese Fragen lösen die Vorgänge 'Prüfen' und 'Vergleichen' aus:

Ein Prüfer aus dem Labor der Überwachungsbehörde prüft zu einem Zeitpunkt die Abwasserprobe nach einer bestimmten Methode für den von ihm abzugebenden Laborbericht; eine Überwachungsbehörde vergleicht danach die Meßwerte aus dem Laborbericht mit den Grenzwerten und stellt folgende Tatbestände fest:

1. Einhaltung der gesetzlichen Normen.
2. Verstoß gegen die gesetzlichen Normen.

Das Ergebnis des Vergleichs zwischen den Meßwerten und Grenzwerten gibt den Umfang und den Inhalt der Problemstellungen in den weiteren Vorgängen vor.

Das Ergebnis des Überwachungsvorganges bestimmt die Problemstellungen in den Bereichen des Abwasservorganges und des Folgevorganges und damit den Umfang und die Reihenfolge der notwendigen Realisierungsstufen von PLIDIS.

2.3. Realisierungsstufen

Die Realisierungsstufen von PLIDIS bestimmen sich nach vom Anwender gesetzten Prioritäten im Bereich der Abwasserüberwachung in Korrelation zu den Anforderungen an das System PLIDIS und den Problemstellungen des Sachbereiches der Abwasserüberwachung.

Die Realisierungsstufen werden in Abbildung 1 dargestellt:

Realisierungs- stufen	Anforderungen an das System Plidis	Problemstellungen des Sachbereiches
	<ul style="list-style-type: none"> -Fragen nach Meßwerten des Probe- entnahmeberichtes und der Labor- untersuchung -Fragen nach dem Vergleich von Meßwerten mit Grenzwerten und mit den vorherigen Meßwerten an den gleichen Meßstellen einer Firma -Fragen nach Probeweltdaten 	<ul style="list-style-type: none"> -Probeentnahmebericht -Laborbericht -Normen (Grenzwerte) -Tatbestand: Einhaltung Verstoß -Firmendaten, Probenehmerdaten, Behörden Daten, Labordaten -Vorgänge: probenehmen/messen, prüfen, auswerten/vergleichen
I	-Fragen nach Betriebsdaten I	<ul style="list-style-type: none"> -Betriebsart, Abwassermenge, Ab- wasserinhaltsstoffe -Vorgänge: produzieren anfallen -Abwasserbehandlung: Kläranlagen Klärverfahren -Vorgänge: reinigen einleiten
II	-Fragen zur Erstellung der Abwasser- bilanz	<ul style="list-style-type: none"> -Anzahl und Ergebnis aller Proben für einen bestimmten Zeitraum -Anzahl und Ergebnis aller Proben eines Probenehmers -Anzahl und Ergebnis aller Proben bei einer Firma -Anzahl und Ergebnis aller Proben eines Wasserwirtschaftsamtes -Anzahl und Ergebnis aller Proben in einem Regierungspräsidium
	-Fragen über Zusammensetzung von Chemikalien (Hilfstoffe)	<ul style="list-style-type: none"> -chemische Hilfstoffe -Verwendung -Giftigkeit -Behandlung -Abbaubarkeit -Ersetzbarkeit -Zusammensetzung
	-Fragen nach den Betriebsdaten II (Erweiterung)	<ul style="list-style-type: none"> -Produktion -Produktionsverfahren -Rohstoffe
	-Fragen nach Abwasserrückverfolgung	<ul style="list-style-type: none"> -geographische Daten: Flußläufe Wasserschutzgebiet Einzugsgebiete Ortschaften -meteorologische Daten -hydrologische Daten

b. 1: Realisierungsstufen von Plidis im Bereich der Abwasserüberwachung

3. Die interne Repräsentation für das Weltmodell 'Abwasserüberwachung'

Einige theoretische Überlegungen zu den Anforderungen an eine geeignete Repräsentationssprache für ein problemlösendes Informationssystem wie PLIDIS werden von ZIFONUN in diesem Band erörtert. Es folgt daraus, daß sich für PLIDIS eine Repräsentationssprache anbietet in der Form einer an der Prädikatenlogik erster Stufe angelehnten Konstruktsprache KS. Diese Sprache vereinigt in sich die Eigenschaften einer Semantiksprache für einen Ausschnitt des Deutschen, einer Modellbeschreibungssprache für die Darstellung des Weltmodells 'Abwasserüberwachung' und einer Datenbasisabfragesprache mit deduktiver Kapazität.

Um diese Anforderungen zu erreichen, waren gewisse Erweiterungen gegenüber dem klassischen logischen Kalkül erforderlich. Diese Erweiterungen liegen hauptsächlich im Bereich der Term-Bildung, der hierarchischen Sortierung der Terme und der Einführung bestimmter 'Quantifikatoren'. Die Syntax und Wohlgeformtheitsbedingungen von KS sind im oben erwähnten Beitrag von ZIFONUN erläutert.

In Zusammenhang mit der Beschreibung des PLIDIS-Systems wird hier lediglich die KS-Einzelsprache für das Weltmodell 'Abwasserüberwachung' skizziert. Diese Skizzierung beschränkt sich damit auf die Festlegung des Vokabulars und des Sortenbaumes für diesen Weltausschnitt.

Der Sortenbaum ist in Abbildung 2 dargestellt. Es zeigt sich, daß dieser nicht von linguistischen Überlegungen her - wie etwa der Darstellung semantischer Merkmale in einem Verbvalenz-Rahmen - motiviert ist, sondern von Überlegungen einer möglichst günstigen Darstellung der Handlungszusammenhänge im Weltausschnitt. Die Sorten können als einstellige Prädikate aufgefaßt werden, deren Extension durch die Menge der Individuen dieser Sorte bestimmt ist. Für jede KS-Individuen-Konstante wird also die Sorte in einem Lexikon eingetragen. Ebenfalls wird für jedes KS-Relations- und Operationszeichen das Sortenwort der Argumente eingetragen. Ein Term kann nur dann als Argument eingesetzt werden, wenn er der im Sortenwort vorausgesetzten oder einer ihr untergeordneten Sorte zugehört.

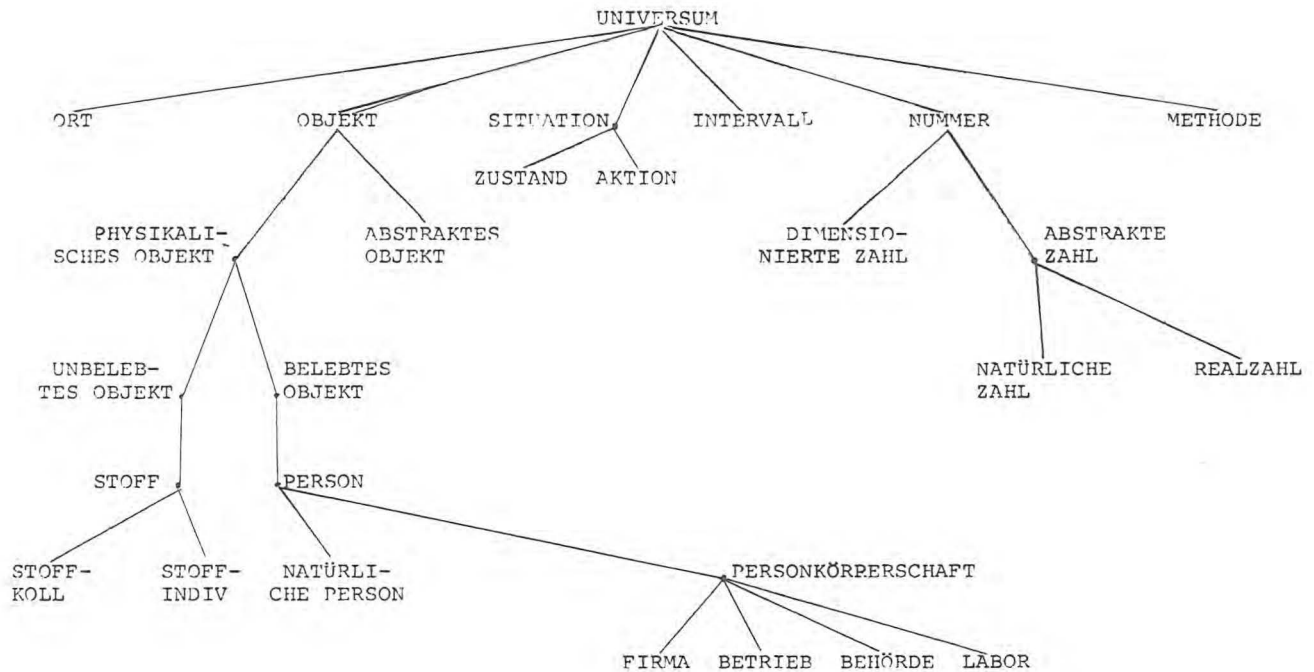


Abbildung 2: Sortenbaum für die KS-Einzelsprache 'Abwasserüberwachung'

Abbildung 3 stellt das wichtigste Vokabular der KS-Einzelsprache 'Abwasserüberwachung' zusammen, und zwar für die Aufgaben in der Ausbaustufe I (vgl. oben Abschnitt 2.3.) zur Beantwortung von Fragen nach Meßdaten und Proben und für die Normenkontrolle.⁵

(a) Beispiele von Individuenkonstanten:

Name	Sorte
ARSEN	STOFF
BRECHT	NATPER
LAUXMANN	FIRMA
PLOCHINGEN	ORT
75/10/13/12/00	INT
0.35-MG/L	DIMZAHL

(b) Relationszeichen

Name	Sortenwort
ADRESSE	<PERS ORT>
IN-LOK	<ORT ORT>
IN-TEMP	<INT INT>
INTER	<INT INT INT>
LETZTNVOR	<NATZAHL (S.J) (S.I) (S.J)>
NACHGEWIESEN	<STOFFK METHODE ABSTROBJ STOFF>
VOR	<INT INT>
ZW	<NATZAHL NATZAHL NATZAHL>

(c) Operationszeichen

Name	Sortenwort
ANTEIL	<STOFF STOFFK ABSTROBJ DIMZAHL>
ANZAHL	<UNI NATZAHL>
BETRIEB	<FIRMA ORT BETRIEB>
GLEICH	<UNI UNI SIT>
GRENZWERTVERLETZUNG	<STOFF SIT>
GROESSER	<NUM NUM SIT>
LABORBERICHT	<LABOR STOFFK INT ABSTROBJ>
MAHNUNG	<STOFFK SIT>
O-GRENZE	<NUM NUM>
PH-WERT	<STOFFK METHODE ABSTROBJ ABSTRZAHL>
PROBE	<BETRIEB INT STOFFK>
PROBE-ENTNAHMEBERICHT	<STOFFK ABSTROBJ>
PROBENEHMER	<STOFFK PER>
TEMPERATUR	<STOFFK INT ABSTROBJ DIMZAHL>
U-GRENZE	<NUM NUM>
VERWARNUNG	<STOFFK SIT>

Abbildung 3: Das wichtigste Vokabular für die KS-Einzelsprache 'Abwasserüberwachung' für die Ausbaustufe I

Mit dieser KS-Einzelsprache werden sowohl das Weltwissen (Fakten und Handlungszusammenhänge) wie faktenspezifische Regeln und Fragen über das Wissen dargestellt. Diese drei Darstellungsmöglichkeiten werden an Hand folgender Beispiele illustriert:

(a) Faktum:⁶

'Die Probe mit der Nummer \$100 ist die Probe vom 13.10.76 um 14 Uhr bei der Firma Guenther Lauxmann, 7144 Asperg, Neckarstraße 7.'

(PROBE (BETRIEB GUENTHER-LAUXMANN 7144/
ASPERG/NECKARSTR/7)
76/10/13/14/00
\$100)

(b) Faktenspezifische Regel:

'Eine Grenzwertüberschreitung eines Meßwertes liegt genau dann vor, wenn dieser Wert größer ist als die gesetzlich zugelassene Obere-Grenze (Norm) für diesen Stoff.'

(FUERALL X.NUM (FUERALL X.SIT
(AEQUIV (GRENZWERTUEBERSCHREITUNG X.NUM X.SIT)
(GROESSER (RESPEKTIV X.NUM)
(O-GRENZE (RESPEKTIV X.NUM))
X.SIT)))

(c) Fragen über das Wissen:

Welches waren die letzten drei Proben der Firma Joos?

(LAMBDA X.STOFFK
(LETZTNVOR 3
(PROBE (BETRIEB JOOS (ORT))(INT))
HEUTE
X.STOFFK))

Bei der Festlegung der Relationen und Operationen einer KS-Einzelsprache müssen einige Anforderungen, resultierend aus ihrer Verwendung als Repräsentationssprache für ein Weltmodell, das als ex-

tensionale Datenbasis realisiert ist, beachtet werden. Sie muß nämlich fakten-orientiert, eindeutig und redundanzfrei sein.

(a) Fakten-orientiert

Bei der Festlegung der Prädikate und ihrer Argumente muß von einer umfassenden Analyse des Weltausschnittes ausgegangen werden, so daß alle zur Erfassung der Handlungszusammenhänge und Sachverhalte notwendigen Merkmale darstellbar sind, und nicht von einer linguistischen Analyse des Sprachgebrauchs. So wurde z.B. zur Darstellung des Sachverhaltes 'Probe' das Prädikat PROBE mit dem Sortenwort <betrieb int stoffkoll> festgelegt. Diese KS-Operation beschreibt also das deutsche *Probe* im Kontext einer bei einem bestimmten Betrieb zu einer bestimmten Zeit genommenen Abwasserprobe, die aus einer Menge von Stoffen ('stoffkoll') besteht und zwar aus solchen Schadstoffen, die im chemischen Labor gemessen werden sollen. Im Kontext Weinprobe wäre selbstverständlich eine ganz andere Darstellung angemessen. Die Angabe eines Zeitarguments für das Prädikat PROBE ergibt sich aus der Tatsache, daß im Weltausschnitt 'Abwasserüberwachung' Proben mittels des Entnahmedatums identifiziert werden, und nicht aufgrund linguistischer Überlegungen. Beim Prädikat BETRIEB war dagegen eine Zeitangabe nicht erforderlich. Wenn sich aber herausgestellt hätte, daß z.B. Angaben über das Gründungsjahr einer Firma eine relevante Information wären, dann müßte das Sortenwort für BETRIEB ein Argument der Sorte 'int' enthalten.

Solche fakten-orientierten Überlegungen motivieren ebenfalls die Festlegung des Sortenbaumes.

(b) Eindeutig

Die Bedingung der Eindeutigkeit ist selbstverständlich, da sonst das Ziehen von falschen Schlüssen nicht auszuschließen wäre. Die Gefahr einer mehrdeutigen Darstellung würde sich z.B. ergeben, wenn eine Argumentensorte nicht genügend spezifiziert wäre.

(c) Redundanzfrei

Die Darstellung einer KS-Einzelsprache soll möglichst vollständig und dazu auch möglichst redundanzfrei sein. Obwohl Redundanzen nicht zu falschen Konsequenzen führen würden, sind sie deshalb zu meiden, da sie einerseits das System undurchsichtig machen und

andererseits zu Unwirtschaftlichkeit führen würden, falls redundante Informationen in der Datenbasis abgespeichert würden.

Es wäre z.B. redundant für die Wiedergabe der deutschen Ausdrücke *die letzten zwei...*, *die letzten drei...*, *die letzten vier...* die Prädikate LETZTZWEIVOR, LETZTDREIVOR, LETZTVIERVOR ... zu definieren. Geeigneter ist es, ein einziges Prädikat LETZTNVOR zu definieren, das ein Argument der Sorte 'natzahl' enthält, das durch die Zahlen 2, 3, 4 ... besetzt werden kann. Dementsprechend kann man auch *der/die/das letzte* ... mit dem Prädikat LETZTNVOR ausdrücken, wobei dann das Argument 'natzahl' durch 1 besetzt ist.

Die Eigenschaften von KS als Abfragesprache werden bei der Beschreibung des Terminiinterpreters erläutert (s. BERRY-ROGGHE/DILGER in diesem Band).

4. Systemarchitektur

4.1. Gesamtüberblick

Abbildung 4 gibt einen Überblick über die Struktur des Systems PLIDIS, über den Informationsfluß zwischen den einzelnen Hauptkomponenten und den Zugriff zu den Datenbasen.

Die Interaktion der Komponenten wird vom PLIDIS-Supervisor gesteuert. Diese Steuerung kann vom Benutzer durch Kommandos in der Kommandosprache beeinflusst werden, was vor allem bei Test- und Wartungsaufgaben notwendig ist.

Der Prozessor für natürlichsprachliche Eingaben analysiert eingebene Sätze und übersetzt sie in die PLIDIS-interne Repräsentationsform (IR).

Zur Verarbeitung von Massendaten stereotyper Form, die nur umständlich in natürlicher Sprache eingegeben werden können, steht ein Prozessor für Formulareingabe zur Verfügung.

Der Prozessor für Information und Problemstellungen muß erkennen, ob es sich bei einer Eingabe um eine abzuspeichernde Information handelt oder um eine Problembeschreibung in Form einer Frage oder eines Befehls. Bei einer Problembeschreibung muß er die notwen-

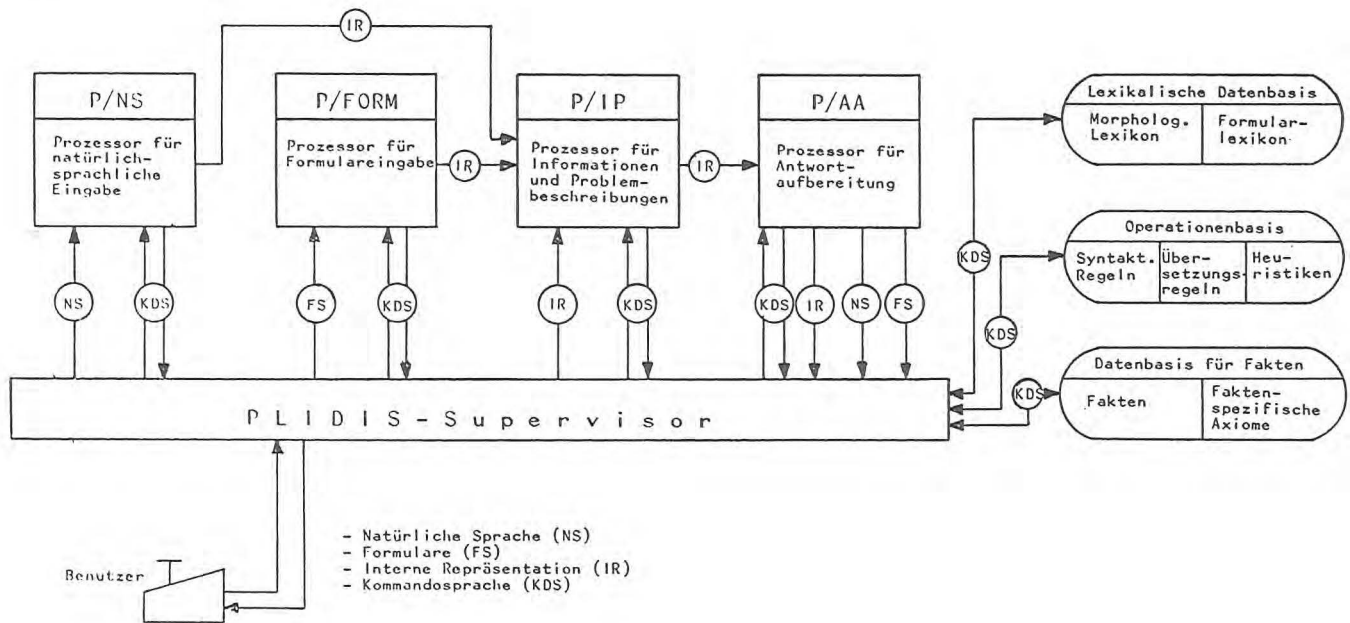


Abbildung 4: PLIDIS-Hauptkomponenten und Datenfluß

(aus: Vorhabenbeschreibung zum Vorhaben PLIDIS, 1977, 13).

gen Daten aus der Datenbasis bereitstellen und eventuell weitere Operationen zur Problemlösung durchführen.

Da die als Antwort auf eine Anfrage ermittelte Information in der internen Repräsentation dargestellt ist, muß die Antwort aus dieser Form in natürliche Sprache zurückübersetzt oder als Tabelle aufbereitet werden. Diese Aufgabe muß ein Prozessor für Antwort-aufbereitung übernehmen.

Die Systemkomponenten greifen entsprechend ihrer Aufgaben auf die Datenbasen zu:

- eine lexikalische Datenbasis, die ein morphologisches Lexikon und ein Formularlexikon umfaßt; das morphologische Lexikon enthält zu jedem verarbeitbaren Wort morphologische Informationen, während im Formularlexikon die Formulare und Tabellen zur formatierten Ein- und Ausgabe verzeichnet sind;
- eine Operationenbasis, welche die syntaktischen Regeln und die Überführungsregeln für eingegebene Äußerungen und Heuristikregeln für den Problemlöser enthält;
- eine Datenbasis im engeren Sinne, welche die Daten und Regularitäten, die für den gewählten Weltausschnitt gelten, enthält. Im konkreten Fall der Abwasserüberwachung bestehen die Fakten aus Massendaten über Laboruntersuchungsberichte, aus Betriebsdaten, aus Daten über Chemikalien und Hilfsstoffe und aus den gesetzlichen Normen für die Abwasserkontrolle. Die faktenspezifischen Gesetzmäßigkeiten beschreiben solche Sachverhalte wie:
eine Verwarnung wird dann ausgesprochen, wenn in der jeweils analysierten Probe und in einer der drei letzten Proben dieser Firma eine Grenzwertverletzung festgestellt wurde.

Oder:

alle Firmen, die im Einzugsgebiet der Rems liegen, liegen auch im Einzugsgebiet des Neckars.

4.2. Die PLIDIS-Hauptkomponenten

Die Konzeptionen für die einzelnen Hauptkomponenten von PLIDIS werden in den nachfolgenden Beiträgen zu diesem Band erläutert. Hier wird nun lediglich ein Überblick gegeben; dabei stehen der konkrete Einsatz und die Leistungsfähigkeit dieser Komponenten im Vordergrund.

Ein Informationssystem umfaßt drei konsekutive Verarbeitungsschritte:

- die Analyse der Eingabe;
- die Informationserschließung bzw. die Informationsabspeicherung;
- die Ausgabe der Antwort.

(1) Die Analyse der Eingabe

In PLIDIS erfolgt die Abfrage von Informationen grundsätzlich in Form einer "normalen" deutschen Frage. Für die Eingabe von Fakten, insbesondere von Massendaten, ist es aber nicht sinnvoll, die natürliche Sprache zu verwenden. Diese Daten sind von der Sache her stereotyp und eignen sich deshalb für eine formatierte Eingabe. Hierzu wird dem Benutzer am Bildschirm ein Formular gezeigt, in das er z.B. nur die in einem Laborbericht enthaltenen Angaben über Meßwerte der untersuchten Stoffe einzutragen hat.

Abbildung 5 zeigt den Zusammenhang der PLIDIS-Komponenten zur Verarbeitung von natürlichsprachlichen Eingabeäußerungen und von Formularen.

Der Prozessor für Formulare wurde mit einem Formulargenerator ausgestattet, der Formularbeschreibungen des Benutzers interpretiert und ein entsprechendes Formular erzeugen kann. Ein Formularübersetzer extrahiert aus einem gegebenen Formular die eingefüllte Information und übersetzt sie in die interne Repräsentationssprache KS. Dabei wird auch überprüft, ob als obligatorisch erklärte Informationen in dem jeweiligen Formular enthalten sind.

Der Prozessor für natürlichsprachliche Eingabe umfaßt drei Komponenten, die die Verarbeitungsschritte eines deutschen Satzes bis zur Überführung in die interne Repräsentation beinhalten:

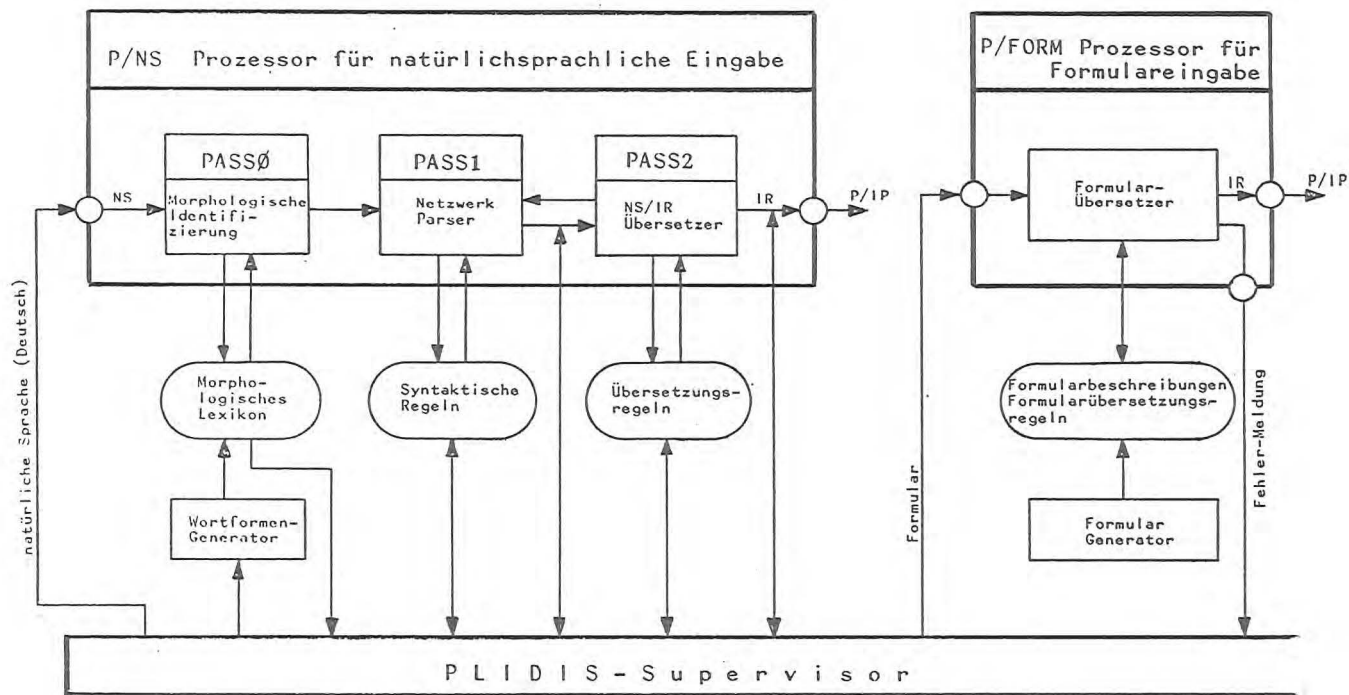


Abbildung 5: PLIDIS-Komponenten zur Verarbeitung von natürlichsprachlicher Eingabe und von Formularen (aus: Vorhabenbeschreibung zum Vorhaben PLIDIS, 1977, 19).

- die morphologische Analyse;
- die syntaktische Analyse;
- die semantische Analyse.

(a) Die morphologische Analyse: PASSØ⁷

Die morphologische Analyse besteht lediglich darin, daß jeder Wortform des Eingabesatzes eine oder mehrere morphosyntaktische Beschreibungen aus einem Vollformen-Lexikon zugeordnet werden. Dieses Verfahren wurde aus Gründen der größeren Geschwindigkeit und der geringeren Fehleranfälligkeit einer morphologischen Analyse mittels eines Stammlexikons vorgezogen. Das morphologische Lexikon enthält den Wortschatz für die Ausbaustufen I und II (vgl. oben 2.3.) und besteht aus etwa 20.000 Einträgen, die in einer externen ISAM-Datei abgespeichert sind. Die Einträge für die geschlossenen Wortklassen des Deutschen sind vollständig. Die "offenen" Klassen umfassen ca. 1.000 Nomina, 400 Verben und 1.000 Adjektive (diese Zahlen beziehen sich jeweils auf die Normalformen, vgl. LÖTSCHER/KOLVENBACH in diesem Band). Um den Aufwand bei der Erstellung eines Vollformen-Lexikons möglichst zu reduzieren, wurden Algorithmen entwickelt, mithilfe derer aus einem gegebenen Lemma automatisch alle flektierten Formen mit den zugehörigen morphosyntaktischen Beschreibungen generiert werden.

(b) Die syntaktische Analyse: PASS1⁸

Die der syntaktischen Analyse zugrundeliegende Grammatik des Deutschen ist in einer Netzwerksprache formuliert. Die Netze werden von einem entsprechenden Parser interpretiert. Das Prinzip entspricht den von WOODS, 1973 beschriebenen "erweiterten Übergangsnetzen" (augmented transition networks), wobei einige für das Deutsche notwendige Anpassungen und Erweiterungen vorgenommen wurden.

Der Grammatiktyp, der mit erweiterten Übergangsnetzen darstellbar ist, geht über die Phrasenstrukturgrammatiken hinaus, da es möglich ist, kontextsensitive Wohlgeformtheitsbedingungen zu formulieren. Die Netzwerksprache bietet dem

Grammatik-Schreiber eine große Flexibilität nicht nur in der Darstellung der Ersetzungsregeln, sondern auch in der Bestimmung von Analysestrategien.

Die für PLIDIS gewählte Analysestrategie ist ein prädikatives top-down-Verfahren.

Als Ergebnis der syntaktischen Analyse wird eine Konstituentenliste des Satzes erzeugt. Es werden dabei keine Abhängigkeitsrelationen zwischen Nominalgruppen angegeben: die Konstituente *bei Lausmann* wird also nicht als Attribut zu *die Proben* im Ausdruck *die Proben bei Lausmann* erkannt. Hierdurch wird vermieden, mehrere Analyseergebnisse für syntaktisch mehrdeutige Konstruktionen zu liefern. Das Erkennen von Attributen wird von der semantischen Analyse geleistet. Wenn ein Satz syntaktisch "falsch" analysiert wurde, wird diese Analyse von der semantischen Komponente verworfen und eine alternative Analyse durch PASS1 vorgenommen.

Die syntaktische Analyse-Komponente kann einen umfangreichen Teil der deutschen grammatischen Strukturen erkennen. Es können folgende Satzstrukturen analysiert werden: Hauptsätze, Relativsätze, adverbiale Nebensätze und Infinitivsätze. Die Nebensätze und Satzglieder können in allen üblichen Satzgliedstellungen stehen. Die Satzdiathese darf sowohl aktiv als auch passiv sein. Der Verbkomplex kann in allen Tempora und in allen Aussageweisen (Indikativ, Imperativ, Konjunktiv) vorkommen. Als Satzkonstituenten werden erkannt: Nominalgruppen, präpositionale Nominalgruppen, Adjektivgruppen, Adverbien und Verbalkomplexe. Die Negation und - mit einigen Einschränkungen - auch die Koordination können analysiert werden.

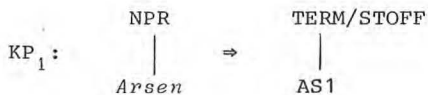
(c) Die semantische Analyse: PASS2⁹

Für die Überführung des syntaktisch analysierten Eingabesatzes in die interne Repräsentationssprache KS wurde eine "Übersetzungsgrammatik" entwickelt. Diese entspricht im Prinzip einer Transformationsgrammatik: Sie operiert über Syntaxbäumen einer Ausgangssprache, die mittels Transformationsregeln verschiedenen Typs in Syntaxbäume der Zielsprache übersetzt werden.

Das Verfahren ist "bottom-up", d.h.: Die Übersetzung beginnt bei den Teilstrukturen aus terminalen und präterminalen Symbolen des Ausgangssprachlichen Syntaxbaumes. So wie bei der syntaktischen Analyse sind die Transformationsregeln von dem interpretierenden Apparat getrennt. Es werden vier Typen von Transformationsregeln unterschieden:

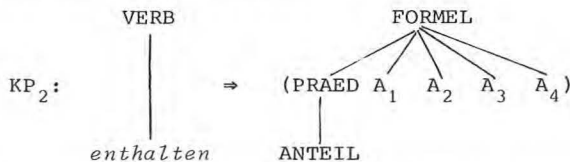
- Kontext-Pattern-Regeln
- Leerstellenregeln
- Translationsregeln
- Terminierungsregeln.

Eine Kontext-Pattern-Regel (CP) bestimmt, durch welche Struktur in der Zielsprache ein terminales Symbol der Ausgangssprache, dominiert von einem bestimmten präterminalen Symbol, übersetzt werden kann. Ein Beispiel einer einfachen CP-Regel wäre:



Diese Regel besagt, daß eine Struktur bestehend aus dem terminalen Symbol ARSEN dominiert von dem präterminalen Symbol NPR durch das KS-Symbol AS1 - einen Term der Sorte 'stoff' - übersetzt werden soll.

Eine etwas kompliziertere Regel definiert die Übersetzung eines Verbs in ein KS-Prädikat:



In der rechten Regelseite wird ein größeres Kontextpattern verwendet, aus dem ersichtlich ist, daß das KS-Symbol ANTEIL als 4-stelliges Prädikat in einer Formel verwendet wird. Die Symbole A_1 , A_2 ... sind Hilfssymbole und können als Markierungen von Leerstellen (slots) verstanden werden, über denen Leerstellenregeln operieren.

Eine Leerstellen- oder Slot-Regel (SL) bestimmt, welche Teilstrukturen für die Ausfüllung der Argumentstellen einer Formel in Frage kommen. Der Syntaxbaum wird also nicht rein linear von links nach rechts abgearbeitet, sondern die Abarbeitung wird von der zu erzeugenden Struktur in der Zielsprache gesteuert. Ein Beispiel einer Leerstellenregel sei die folgende Regel:

SL: $A_3 \Rightarrow \text{TERM, ABSTROBJ}$
 (PATTERN (PNG (ODER (PRAEP DAT
 LAUT ENTSPRECHEND
 NACH GEMAESS)
 (PRAEP GEN AUFGRUND))))

In dieser Regel wird festgelegt, daß das Hilfssymbol A_3 (der oben angeführten ANTEIL-Formel) durch die von einem Term der Sorte 'abstraktes Objekt' gebildete Struktur ersetzt werden soll. Diese Struktur ist auffindbar als Übersetzung einer präpositionalen Nominalgruppe im Dativ, die mit einer der Präpositionen *laut*, *entsprechend*, *nach*, *gemäß* beginnt, oder einer präpositionalen Nominalgruppe im Genitiv, die mit der Präposition *aufgrund* beginnt und jeweils ein Nomen, dessen Translat von der Sorte 'abstrobj' ist, enthält. Einer solchen Teilstruktur würde zum Beispiel der Ausdruck *laut Laborbericht* entsprechen.

Die Translate, die durch Anwendung von KP- und SL-Regeln erzeugt wurden, werden an den entsprechenden Stellen im Syntaxbaum der Ausgangssprache eingeordnet. Da dies in der Regel an den unteren Knoten geschieht, müssen Translathebens- oder Pattern-Raising-Regeln (PR) formuliert werden, die die Bedingungen definieren, unter denen gewisse Translat-Gebilde im Baum "nach oben" weitergegeben werden können.

Terminierungsregeln schließlich sind ein besonderer Typ von PR-Regeln, die nur auf die Wurzel S (den obersten Knoten im Syntaxbaum) angewandt werden können.

Das Ergebnis der Überführung besteht dann aus der Formel, die sich ergibt aus der linearen Anordnung der terminalen Symbole derjenigen Translat-Struktur, die dem Knoten S zugeordnet wurde.

(2) Die Informationserschließung und -abspeicherung

Abbildung 6 zeigt die Struktur der PLIDIS-Komponente zur Informationsverarbeitung und Problemlösung.

Der Monitor prüft, ob die gegebene KS-Formel eine faktische Aussage bzw. eine faktenspezifische Regel oder eine Frage über das Weltwissen darstellt. Im ersten Fall wird sie zur Abspeicherung an die Datenbasisverwaltung weitergegeben. Im zweiten Fall wird sie der Problemlösungskomponente übergeben. Letztere enthält einen Theorembeweiser nach dem Resolutionprinzip, der im Stande ist, über Formeln der Prädikatenlogik erster Stufe zu deduzieren. Die hierfür bekannten Algorithmen wurden so modifiziert, daß sie die Sortenstruktur von KS berücksichtigen. Die Sorten schränken die Zahl der möglichen Unifikatoren ein, was den Deduktionsvorgang beschleunigt. Der Theorembeweiser kann beliebige syntaktisch korrekte Formeln verarbeiten, ohne auf den Inhalt der Formeln Bezug zu nehmen. Hiermit ist die Übertragbarkeit dieser Komponente auf andere Weltausschnitte gegeben. Es werden zwar bei der Deduktion faktenspezifische Axiome herangezogen - und außerdem können faktenspezifische Hinweise (Heuristiken) den Deduktionsvorgang steuern -; diese werden aber rein syntaktisch verwendet.

Die Steuerkomponente des Problemlösungsapparats ist der Termininterpreter, der die KS-Fragen interpretiert, d.h. die erforderlichen Datenbasisaufrufe erzeugt, Operationen anstößt und den Theorembeweiser gegebenenfalls zur Deduktion heranzieht. Die Arbeitsweise des Termininterpreters wird anhand eines Beispiels im nächsten Abschnitt erläutert.¹⁰

Schließlich umfaßt die Komponente noch zusätzlich auf den Anwendungsbereich zugeschnittene Operationen sowie statistische Berechnungen, die ebenfalls vom Einsatzbereich von PLIDIS her motiviert sind.

Logisch gesehen repräsentiert die PLIDIS-Datenbasis ein extensionales Modell einer KS-Einzelsprache; d.h.: Sie besteht nur aus atomaren KS-Formeln, die keine Variablen enthalten. Unter dem Datenbank-Aspekt kann sie als relationale Datenbasis aufgefaßt werden (s. CODD, 1970).

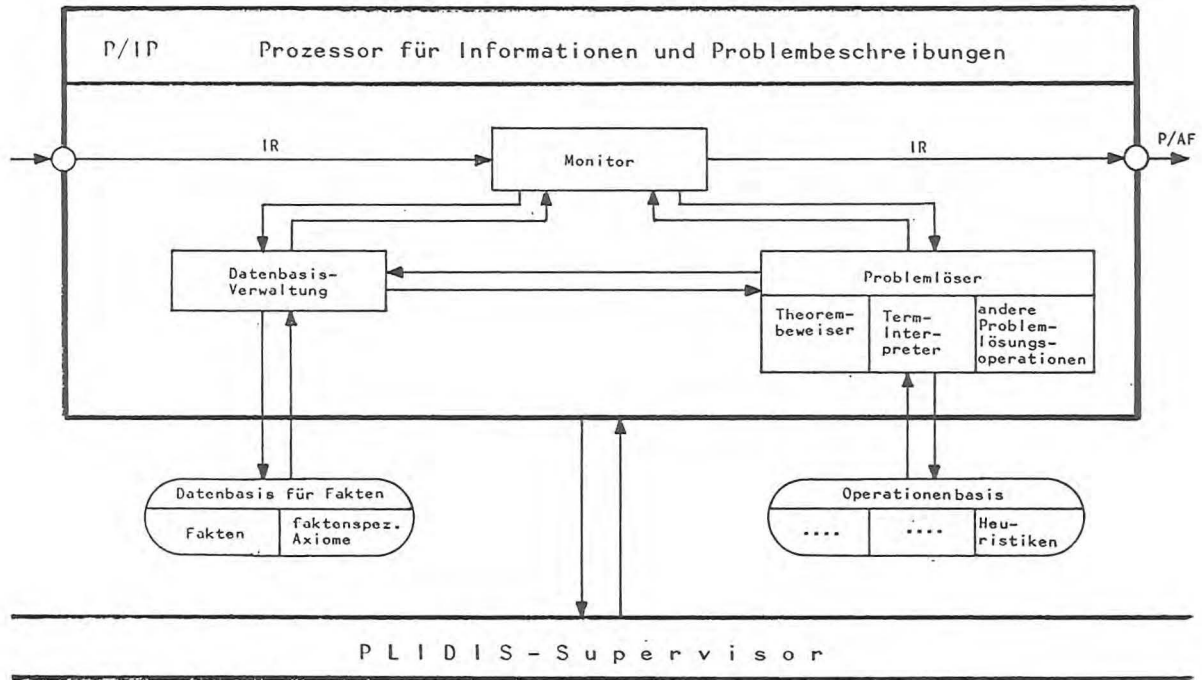


Abbildung 6: Struktur der PLIDIS-Komponente für die Informationsverarbeitung und Problemlösung (aus: Vorhabenbeschreibung zum Vorhaben PLIDIS, 1977, 21)

Intern ist die Datenbasis in Form von zwei Dateien, einer Primärdatei und einer Sekundärdatei realisiert. Die erstere enthält atomare Formeln als Daten, die zweite alle vorkommenden Symbole zusammen mit Pointern zu den Daten der Primärdatei, in denen die Symbole vorkommen.

Es ist die Aufgabe der Datenbasisverwaltung, die Daten in einer der Struktur der Datenbasis angepaßten Form abzuspeichern und auf Anforderung wieder zur Verfügung zu stellen. Bei der Abspeicherung muß auch eine Konsistenzprüfung durchgeführt werden, damit keine sich gegenseitig widersprechenden oder duplizierten Informationen abgespeichert werden. Jedes abzuspeichernde Datum wird mit einem Sicherheitsschlüssel versehen. Es wurde eine Hierarchie der Zugriffsberechtigung festgelegt, so daß jeder Benutzer nur auf solche Daten zugreifen kann, die von ihm selbst oder von in der Hierarchie unter ihm stehenden Benutzern eingegeben worden sind.

(3) Die Ausgabe der Antwort

Falls ein positives Ergebnis zu einer Einsetzungsfrage vorliegt, werden die als Antwort gelieferten KS-Formeln in der Antwortaufbereitungskomponente tabellarisch aufbereitet, damit sie vom Benutzer leichter gelesen werden können.

Für bestimmte Fragen nach Handlungsvorgängen (warum- und wie-Fragen) wäre eine natürlichsprachliche Ausgabe angemessen. Im derzeitigen PLIDIS-System ist eine Komponente, die aus KS-Formeln deutsche Sätze erzeugt, noch nicht vorhanden, ihre Erstellung ist aber geplant.

Bei einer positiv beantworteten Entscheidungsfrage werden außer der Antwort JA noch zusätzliche Informationen ausgegeben. Z.B. werden auf die Frage *Wurde die Firma Lauxmann im Jahr 1976 überprüft?* alle Probeentnahmedaten ausgegeben.

Wenn eine Frage nicht beantwortet werden konnte, kann dies verschiedene Ursachen haben:

- Die Person, die die Frage gestellt hat, hat keine Zugriffsberechtigung auf die angeforderten Daten.
- Eine in der Frage enthaltene Präsupposition stellt sich als falsch heraus.

- Das Prädikat einer Abfrage ist nicht in der Datenbasis enthalten und es besteht auch kein Axiom über dieses Prädikat.
- Eine Datenbasisabfrage kann nicht erfüllt werden (im Falle einer Einsetzungsfrage).

Für alle diese Fälle werden entsprechende Systemmeldungen erzeugt.

5. Beispielanalyse einer Frage an PLIDIS

Anhand eines Protokolls, in dem eine Interaktion mit dem System PLIDIS im Zusammenhang festgehalten ist, sollen im folgenden die einzelnen Verarbeitungsschritte und ihre Ergebnisse demonstriert und erläutert werden ("(IN)" am linken Rand des Protokollausdrucks signalisiert, daß es sich um eine Eingabe durch den Benutzer handelt, "(OUT)" kennzeichnet die vom System erzeugte Ausgabe). Vgl. das Protokoll S. 68ff.

Nachdem PLIDIS aufgerufen wurde, prüft das System zunächst die Zugriffsberechtigung des Benutzers mittels eines Passwortes und einer zu errechnenden Zahl.

Damit die Zwischenergebnisse der morphologischen und syntaktischen Analyse mitprotokolliert werden, werden die folgenden PLIDIS-Systemkommandos eingegeben:

/AUSGABE PASSØ EIN

/AUSGABE PASS2 EIN

Das weitere Kommando

/TRACE PASS1 EIN

bewirkt, daß der Parsing-Ablauf durch das erweiterte Übergangsnetz in PASS1 noch zusätzlich aufgelistet wird.

Die Beispielfrage lautet:

Welche Proben der Firma Lausmann haben 1975 Grenzwerte überschritten?

```

(IN)      (PLIDIS)
(OUT)
(OUT)      WILLKOMMEN IN PLIDIS
(OUT)
(OUT)
(OUT)      (VERSION 2.1 - OKTOBER 1977)
(OUT)
(OUT)
(OUT)      BITTE PASSWORD EINGEBEN:
(OUT)
(IN)      ZIMPEL
(OUT)
(OUT)      BITTE Y FÜR X=13543 EINGEBEN
(OUT)
(IN)      13543
(OUT)
(OUT)      PLIDIS IST INTERAKTIONSBEREIT
(OUT)
(IN)      /AUSGABE PASS0 EIN
(IN)      /AUSGABE PASS1 EIN
(IN)      /TRACE PASS1 EIN
(OUT)
(OUT)      PLIDIS IST INTERAKTIONSBEREIT
(OUT)
(IN)      WELCHE PROREN VON LAUXMANN HABEN 1975/ GRENZWERTE
        UEBERSCHRITTEN ?
(OUT)
(OUT)
(OUT)      ERGEBNIS DER MORPHOLOGISCHEN ANALYSE (PASSE):
(OUT)
(OUT)      (WELCHE (WDET NF WELCH KNG 4682 K
(OUT)      (NOM AKK)
(OUT)      PN
(OUT)      (3 6)
(OUT)      G F F STARK FF SCHWACH)
(OUT)      (WDET NF WELCH KNG 4620 K
(OUT)      (NOM AKK)
(OUT)      PN 6 G M F STARK FF SCHWACH)
(OUT)      (WDET NF WELCH KNG 4617 K
(OUT)      (NOM AKK)
(OUT)      PN 6 G M F STARK FF SCHWACH)
(OUT)      (RELPRON NF RELWELCHER KNG 4682 K
(OUT)      (NOM AKK)
(OUT)      PN
(OUT)      (3 6)
(OUT)      G F)

```

(OUT) (PELPRON NF RELWELCHER KNG 4620 K
 (OUT) (NOM AKK)
 (OUT) PN 6 G M)
 (OUT) (REIPRON NF RELWELCHER KNG 4617 K
 (OUT) (NOM AKK)
 (OUT) PN 6 G N)
 (OUT) (WPRON NF WELCHER KNG 4682 K
 (OUT) (NOM AKK)
 (OUT) PN
 (OUT) (3 6)
 (OUT) G F)
 (OUT) (WFRON NF WELCHER KNG 4620 K
 (OUT) (NOM AKK)
 (OUT) PN 6 G M)
 (OUT) (WPRON NF WELCHER KNG 4617 K
 (OUT) (NOM AKK)
 (OUT) PN 6 G N))
 (OUT) (PROHEN (H NF PROBE KNG 7690 K
 (OUT) (NOM GEN DAT AKK)
 (OUT) PN 6 G F)
 (OUT) (VERB NF PROHEN PN (2 5)
 (OUT) MOD BEF TEMP GE DIA AKT)
 (OUT) (VERB NF PROHEN TEMP IN)
 (OUT) (VERB NF PROHEN PN (4 6)
 (OUT) MOD
 (OUT) (IND KONJ)
 (OUT) TEMP GE DIA AKT)
 (OUT) (VERB NF PROHEN KNG 7831 PN
 (OUT) (2 5)
 (OUT) MOD BEF TEMP GE DIA AKT)
 (OUT) (VERB NF PROHEN KNG 8191 TEMP IN)
 (OUT) (VERB NF PROHEN KNG 7727 PN
 (OUT) (4 6)
 (OUT) MOD
 (OUT) (IND KONJ)
 (OUT) TEMP GE DIA AKT))
 (OUT) (VON (PRAEP NF VON KNG 1535 R DAT))
 (OUT) (LAUXMANN (NPR NF LAUXMANN KNG 7750 K
 (OUT) (NOM GEN DAT AKK)
 (OUT) PN 3 G (M F)))
 (OUT) (HABEN (VERB NF HABEN KNG 7831 PN
 (OUT) (2 5)
 (OUT) MOD BEF TEMP GE DIA AKT)
 (OUT) (VERB NF HABEN KNG 7727 PN
 (OUT) (4 6)
 (OUT) MOD
 (OUT) (IND KONJ)
 (OUT) TEMP GE DIA AKT INF 2)
 (OUT) (VERB NF HABEN KNG 8191 TEMP IN INF 2)
 (OUT) (AUXH NF HABEN KNG 7727 PN
 (OUT) (4 6)
 (OUT) MOD
 (OUT) (IND KONJ)
 (OUT) TEMP GE DIA AKT INF 4)

```

(OUT)          (AUXH NF HABEN KNG 8191 TEMP IN INF 4))
(OUT)          (1975/ (DATUMSZAHL NF 75/ KNG 7748))
(OUT)          (GRENZWERTE (N NF GRENZWERT KNG 1092 K DAT PN
(OUT)          3 G M)
(OUT)          (N NF GRENZWERT KNG 6668 K
(OUT)          (NOM GEN AKK)
(OUT)          PN 6 G M))
(OUT)          (UEBERSCHRITTEN (VERB NF UEBERSCHREITEN KNG
(OUT)          7727 PN (4 6)
(OUT)          MOD
(OUT)          (IND KOMJ)
(OUT)          TEMP VE DIA AKT)
(OUT)          (VERB NF UEBERSCHREITEN KNG
(OUT)          8191 TEMP P2 HS HABEN
(OUT)          PAS 1)
(OUT)          (ADJU NF UEBERSCHRITTEN KNG
(OUT)          8191 S POS))
(OUT)          (?)
(OUT)          (INTERP NF ? TERMINAL T TYPE FRAGE))
(OUT)
(OUT)          PASSO          CPU: 4.244 SEC          REAL: 0:36 MIN
(OUT)
(OUT)          PUSH S/
(OUT)          PUSH KG/
(OUT)          MCAT WDET GEFUNDEN /WELCHE/
(OUT)          STRING = (PROBEN VON LAUXMANN HABEN 1975/
(OUT)          GRENZWERTE UEBERSCHRITTEN ?)
(OUT)          GO TO STATE NP/DET
(OUT)          MCAT N GEFUNDEN /PROBEN/
(OUT)          STRING = (VON LAUXMANN HABEN 1975/
(OUT)          GRENZWERTE UEBERSCHRITTEN ?)
(OUT)          GO TO STATE NP/NP
(OUT)          ERGEBNIS VON POP /(WELCHE PROBEN)/ :
(OUT)          (NG ((KNG . 4618) (NS WELCHE PROBEN)) (WDET
(OUT)          NIL WELCH) (N NIL PROBE))
(OUT)          GO TO STATE S/IK
(OUT)          PUSH PNG/
(OUT)          CAT PRAEP GEFUNDEN /VON/
(OUT)          STRING = (LAUXMANN HABEN 1975/ GRENZWERTE
(OUT)          UEBERSCHRITTEN ?)
(OUT)          GO TO STATE NP/PRAEP
(OUT)          JUMP TO STATE NP/DET
(OUT)          PUSH ZAHL/
(OUT)          JUMP TO STATE ZAHL/ZAHLADV
(OUT)          WEG GESPERRT
(OUT)          WEG GESPERRT
(OUT)          CAT NPR GEFUNDEN /LAUXMANN/
(OUT)          STRING = (HABEN 1975/ GRENZWERTE
(OUT)          UEBERSCHRITTEN ?)
(OUT)          GO TO STATE NP/NP
(OUT)          ERGEBNIS VON POP /(VON LAUXMANN)/ :
(OUT)          (PNG ((KNG . 1794) (NS VON LAUXMANN)) (PRAEP
(OUT)          NIL VON) (NPR NIL LAUXMANN))

```



```

(OUT)          CAT VERB GEFUNDEN /UEBERSCHRITTEN/
(OUT)          STRING = (?)
(OUT)          GO TO STATE HSVK/P2
(OUT)          CAT AUXH IN STACK VFIN GEFUNDEN /HABEN/
(OUT)          HOLDSTACK(VFIN) = NIL
(OUT)          GO TO STATE HSVK/VK
(OUT)          ERGEBNIS VON POP /UEBERSCHRITTEN/ :
(OUT)          (VK ((PN 4 6)) (V ((TEMP . PFT)) UEBERSCHREITEN))
(OUT)          GO TO STATE S/VERB
(OUT)          WORD ? GEFUNDEN
(OUT)          STRING = NIL
(OUT)          GO TO STATE S/S
(OUT)          ERGEBNIS VON POP /(WELCHE PROBEN VON
(OUT)          LAUXMANN HABEN 1975/ GRENZWERTE UEBERSCHRITTEN
(OUT)          ?)/ :
(OUT)
(OUT)          ERGEBNIS DER MORPHO-SYNTAKTISCHEN ANALYSE (PASS1):
(OUT)
(OUT)
(OUT)          (S ((TYPE . FRAGE)
(OUT)              (DIATHESE . AKTIV)
(OUT)              (NS . (WELCHE PROBEN VON LAUXMANN
(OUT)              HABEN 1975/ GRENZWERTE UEBERSCHRITTEN ?)))
(OUT)              (VK ((PN . (4 6)))
(OUT)                  (V ((TEMP . PFT))
(OUT)                      UEBERSCHREITEN))
(OUT)              (NG ((KNG . 4618)
(OUT)                  (NS . (WELCHE PROBEN)))
(OUT)                  (WDET NIL
(OUT)                      WELCH)
(OUT)                  (N NIL
(OUT)                      PROBE))
(OUT)              (PNG ((KNG . 1094)
(OUT)                  (NS . (VON LAUXMANN)))
(OUT)                  (PRAEP NIL
(OUT)                      VON)
(OUT)                  (NPR NIL
(OUT)                      LAUXMANN))
(OUT)              (VERB ((NS . HABEN))
(OUT)                  HABEN)
(OUT)              (NG ((KNG . 7748)
(OUT)                  (NS . 1975/))
(OUT)                  (DATUMSZAHL NIL
(OUT)                      75/))
(OUT)              (NG ((KNG . 1392)
(OUT)                  (NS . GRENZWERTE))
(OUT)                  (N NIL
(OUT)                      GRENZWERT))
(OUT)              (VERB ((NS . UEBERSCHRITTEN))
(OUT)              ))
(OUT)
(OUT)          PASS1          CPU: 1.774 SEC          REAL: 0:38 MIN
(OUT)

```

```

(OUT)
(OUT)
(OUT)
(OUT)
(OUT)   ERGEBNIS DER KS-UEBERFUEHRUNG (PASS2):
(OUT)
(OUT)   <LAMBDA X2
(OUT)   (EXIST X1.SIT
(OUT)   (EXIST X3.ORT
(OUT)   (EXIST X4.STOFF
(OUT)   (EXIST X5.AOB
(OUT)   (UND (PROBE(BETRIEB G-L X3.ORT)(IN-TEMP 75/)X2)
(OUT)   (GRENZWERTUEBERSCHREITUNG
(OUT)   (ANTEIL X4.STOFF X2 X5.AOB)X1.SIT>
(OUT)
(OUT)   PASS2           CPU: 0.910 SEC      REAL: 0:09 MIN
(OUT)
(OUT)
(OUT)   PROBE           BETRIEB
(OUT)
(OUT)   75/11/29/14/00   GUENTER-LAUXMANN
(OUT)   75/07/09/14/30   GUENTER-LAUXMANN
(OUT)   75/02/27/13/00   GUENTER-LAUXMANN
(OUT)
(OUT)   PASS3           CPU: 51.746 SEC      REAL: 2:47 MIN
(OUT)   TOTAL           CPU: 58.782 SEC      REAL: 4:14 MIN
(OUT)
(OUT)   FLIDIS IST INTERAKTIONSBEREIT
(OUT)
(IN)   /ENDE

```

5.1. Die morphologische Analyse

Das Ergebnis der morphologischen Analyse ist die Zuordnung der entsprechenden Lexikoneinträge zu den einzelnen Wörtern des Eingabesatzes. Die morphologische Mehrdeutigkeit mancher deutscher Wörter ist bekannt. Dem Wort *Proben* z.B. können folgende morphosyntaktischen Beschreibungen zugeordnet werden:

- Nomen: Feminin Plural, im Nominativ, Genitiv, Dativ
oder Akkusativ
- Verb: im Imperativ, Gegenwart, zweite Person Singular oder zweite Person Plural (Höflichkeitsform)
- Verb: im Infinitiv
- Verb: im Indikativ oder Konjunktiv, Gegenwart, erste oder dritte Person Plural.

Die Auflösung dieser Mehrdeutigkeiten findet, soweit sie aufgrund morphosyntaktischer Regularitäten möglich ist, im nächsten Verarbeitungsschritt statt.

5.2. Die syntaktische Analyse

Es ist die Aufgabe der syntaktischen Analyse, die Satzkonstituenten zu erkennen und einen Syntaxbaum zu erzeugen. Der Ablauf dieser Analyse durch das Netzwerk kann mit Hilfe des Trace-Protokolls verfolgt werden.

Das "top-level"-Netz, nämlich das "Satz-Netz", beginnt mit dem Zustand S/. Es wird zunächst das "Nominal-Gruppen-Netz" (NG/) angestoßen. Da das Wort *welche* das Merkmal WDET aufweist, kann diese Kante begangen werden, und das Wort wird aus dem Eingabestring entfernt.

Vom Zustand NP/DET kann zum Zustand NP/NP übergegangen werden, da das nächste Wort des Eingabesatzes (*Probe*) zur Kategorie Nomen gehört.

Hiermit ist die erste Konstituente "Nominalgruppe" erkannt, und die POP-Kante besagt, daß das NG-Netz verlassen wird. Beim Verlassen dieses Netzes wird als Teilergebnis der Analyse folgende Information abgespeichert:

(NG ((KNG . 4618)(NS WELCHE PROBEN)) (WDET NIL WELCH) (N NIL PROBE)). Diese Angabe beschreibt die Konstituente als eine Nominalgruppe (NG), die aus einem Fragedeterminans (WDET) und einem Nomen (N) besteht. Das Merkmal KNG ist eine Verschlüsselung für das Bit-Muster, das die Merkmale Kasus, Numerus und Genus darstellt. Die Zahl 4618 entspricht den Merkmalen: Feminin, Plural, Nominativ oder Akkusativ. Im Hauptnetz erfolgt nun zunächst der Übergang zum Zustand S/NK. Von hier aus wird das Unternetz "Präpositionale Nominalgruppe" (PNG/) angestoßen. Nachdem *von* (als nächstes im Eingabesatz zu analysierendes Wort) als Präposition erkannt wurde, kann zum Zustand NP/PRAEP übergegangen werden. Es wird zunächst geprüft, ob nach der Präposition ein Determinans, ein Pronomen oder das Wort *wieviele* vorliegt. Da dies in dem vorliegenden Eingabesatz nicht der Fall ist, kann über eine JUMP-Kante der Zustand NP/DET erreicht werden. Das Unternetz ZAHL/ wird erfolglos angestoßen. Zuletzt wird geprüft, ob ein Eigennamen vorliegt. Da dies der Fall ist, wird die Konstituente *von Lausmann* erfolgreich als eine Präpositionale Nominalphrase analysiert.

Es wird zum Zustand S/NK im Hauptnetz zurückgekehrt. Das nächste zu analysierende Wort im Eingabestring, nämlich *haben*, wird als Verb erkannt und aus dem String entfernt. Vom Zustand S/NK wird nun zum Zustand S/VFIN übergegangen. Bei diesem Zustand wird geprüft, ob das nächste Wort ein Reflexivpronomen ist; da dies nicht der Fall ist, geschieht ein folgenloser Übergang (JUMP) zum Zustand S/VK.

Nachdem zuerst erfolglos das PNG-Netz angestoßen wird, wird das Wort *1975/* als Datumszahl erkannt, die für sich eine NG-Konstituente darstellt.

Das Wort *Grenzwerte* wird gleichfalls als NG analysiert. Die korrekte Bestimmung des letzten im Eingabestring verbliebenen Wortes *überschritten* ist ein etwas langwieriger Prozeß.

Die Bearbeitung der PUSH-Kante zur PNG wird gleich abgebrochen. Die nächste PUSH-Kante zur NG hat zunächst Erfolg, da *überschritten* aufgrund seiner morphologischen Bestimmung zuerst als Adjektiv erkannt wird. Dieser Schritt muß aber rückgängig gemacht werden, da auf das unflektierte Adjektiv kein flektiertes Adjektiv folgt, wie dies vom ADJG/-Netz gefordert wird.

Wieder auf "top-level" werden der Reihe nach die "Relativsatz"- und "Artangabe-Netze" erfolglos durchgegangen. Zuletzt wird das "Hauptsatzverbkomplex-Netz" (HSVK/) angestoßen. Vom Zustand HSVK/ gehen verschiedene CAT-Kanten aus, die die Kategorie Verb voraussetzen. Dies trifft zwar auf *überschritten* zu, aber die Kontextbedingungen sind in den ersten drei Fällen nicht erfüllt: das Verb müßte dabei nämlich im Infinitiv stehen. Die Testbedingung, daß das Verb im Partizip Perfekt stehen muß, wird im vierten Versuch erfüllt, und nun kann zum Zustand HSVK/P2 übergegangen werden. Die nächste begehbare Kante ist eine VIR-Kante, die die Behandlung von diskontinuierlichen Elementen erlaubt. Nachdem das Hilfsverb *haben* im Holdstack aufgefunden wurde, wird die Verbkonstituente erkannt und das "top-level-Netz" erfolgreich verlassen. Das Analyse-Ergebnis von PASS1 wird - in vergrößerter Form - in Abbildung 7 graphisch dargestellt.

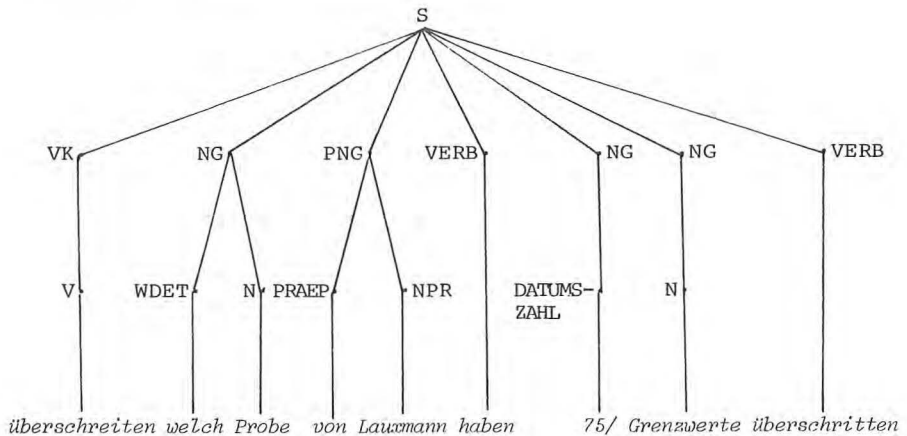


Abbildung 7: Syntaxbaum für den Satz: *Welche Proben von Lauxmann haben 1975 Grenzwerte überschritten?*

Der Satz *Welche Proben von Lauxmann haben 1975/ Grenzwerte überschritten.* besteht demnach aus den folgenden Hauptkonstituenten: VK - NG - PNG - VERB - NG - NG - VERB. Daß die PNG-Konstituente im Falle des vorliegenden Satzes zur vorangehenden NG-Konstituente in einer Attribut-Beziehung steht, wird nicht erkannt. Diese Beziehung wird im nächsten Verarbeitungsschritt, im PASS2, nachgewiesen.

5.3. Die Überführung in KS

Der Überführungsprozeß, der in PASS2 erfolgt, besteht darin, mit Hilfe von Transformationsregeln aus dem Syntaxbaum, der in PASS1 für den eingegebenen Fragesatz erzeugt worden ist, die folgende KS-Formel zu generieren:

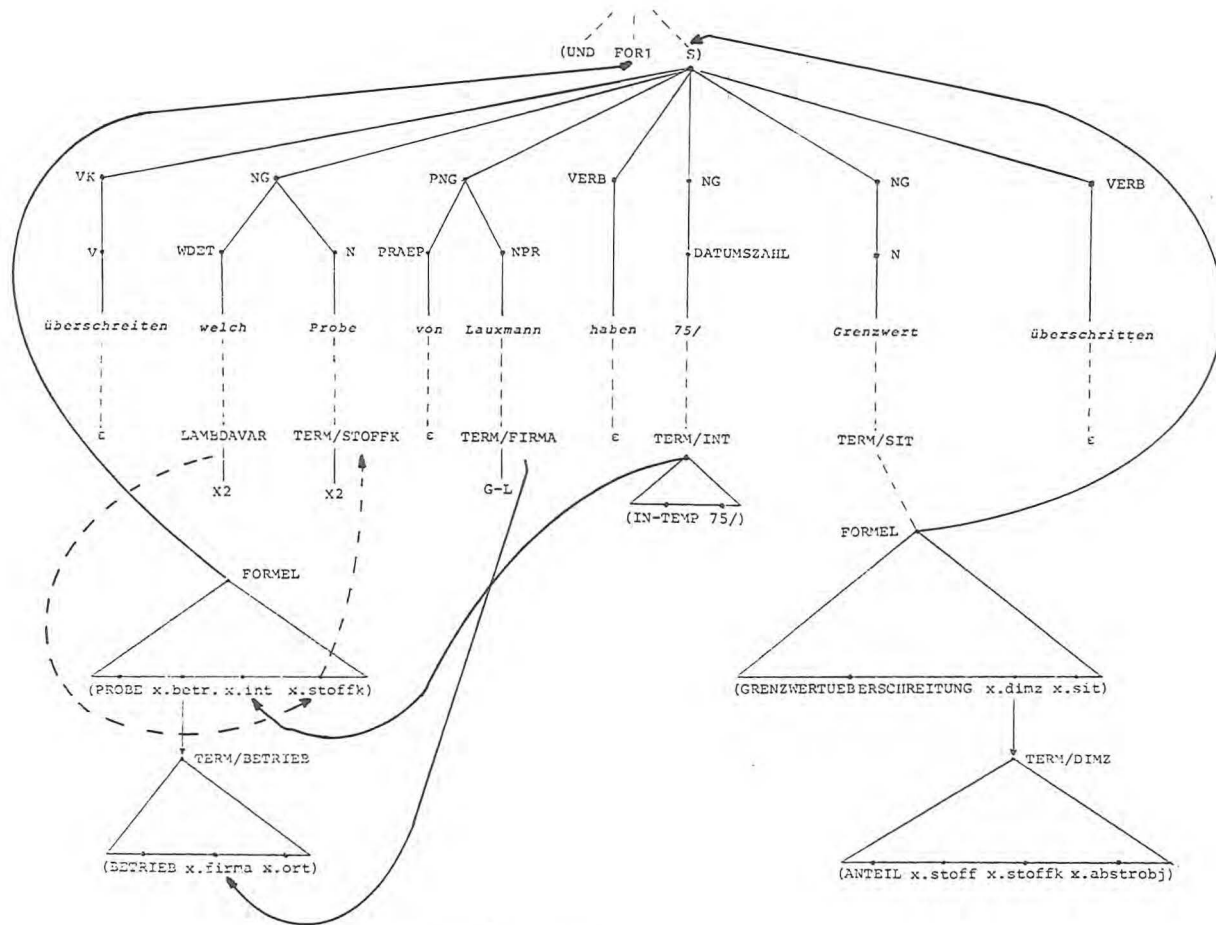
```
(LAMBDA X2
  (EXIST X1.SIT
    (EXIST X3.ORT
      (EXIST X4.STOFF
        (EXIST X5.AOB
          (UND (PROBE (BETRIEB G-L X3.ORT) (IN-TEMP 75/) X2)
            (GRENZWERTUEBERSCHREITUNG
              (ANTEIL X4.STOFF X2 X5.AOB) X1.SIT)))))))))
```

In Abbildung 8 wird versucht, mit entsprechenden Vereinfachungen das Verfahren graphisch darzustellen.

Unter jedes der terminalen Elemente des Syntaxbaumes (vgl. Abbildung 7) wird das entsprechende KS-Translat angehängt. (ε bedeutet, daß für ein Symbol keine unmittelbare Übersetzung in der Zielsprache besteht, was jedoch nicht bedeutet, daß die Information des Symbols nicht berücksichtigt wurde.)

Aus Gründen der Übersichtlichkeit ist die Auswirkung der Regeln zur Translathebung, die definieren, wie die Translate "nach oben" zum S-Knoten gebracht werden, nicht eingezeichnet.

Der Überführungsprozeß startet beim ersten terminalen Symbol des natürlichsprachlichen Syntaxbaumes, bei *überschreiten*. Im Rahmen



der Fügung "einen Grenzwert überschreiten" wird das Verb *überschreiten* mit der leeren Kette ϵ übersetzt. Denn das Verb hat keine direkte Entsprechung in KS; die mit *überschreiten* formulierte Information wird jedoch bei der Übersetzung von *Grenzwert* berücksichtigt. Damit wird auch den Knoten mit V und VK als Translat ϵ zugeordnet.

Als nächstes Symbol wird *welch* übersetzt. Die Überführungsregel für *welch* erzeugt als Translat eine Variable X2 mit der Kennzeichnung LAMBDABAR. D.h.: Wird die Variable als Argument in eine Formel eingesetzt, muß die Formel am Ende des Überführungsvorgangs durch Präfigierung mit LAMBDA X2 zu einem LAMBDA-Abstrakt umgeformt werden.

Das nächste Symbol, *Probe*, erzeugt als Translat einen Term der Sorte "stoffkoll" mit dem Prädikat PROBE, das als Argumente das Sortenwort $\langle x.betr \ x.int \rangle$ nimmt. Dies wird ausgedrückt durch das Pattern für einen Ausdruck mit PROBE als Formel: $(PROBE \ x.betr \ x.int \ x.stoffk)$, wobei *x.betr*, *x.int*, *x.stoffk* als Leerstellen (slots) des Pattern aufzufassen sind und durch Translate aus dem natürlichsprachlichen Kontext von *Probe* auszufüllen sind.

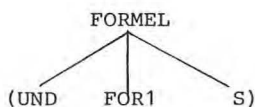
Die Leerstellenregel für *x.betr* legt u.a. folgende Alternativen fest:

- (1) *x.betr* kann ersetzt werden durch das Translat einer NG oder PNG, die in der syntaktischen Relation ATTRIBUT zu *Probe* steht; das Translat muß ein Term der Sorte "betrieb" sein;
- (2) *x.betr* kann ersetzt werden durch ein Pattern (BETRIEB *x.firma x.ort*); durch Ausfüllung der Leerstelle *x.firma* durch ein Translat der Sorte "firma" eines Attributs von *Probe* und der Leerstelle *x.ort* durch ein Translat der Sorte "ort" eines weiteren Attributs von *Probe* entsteht dann ein Term der Sorte "betrieb";
- (3) *x.betr* kann, falls die beiden vorangehenden Möglichkeiten nicht realisierbar sind, durch eine Variable *x* mit der Kennzeichnung EXISTVAR ausgefüllt werden, die am Ende der Überführung, ähnlich wie bei LAMBDABAR, durch einen Existenzquantor zu binden ist.

Da (1) im Beispielsatz nicht realisierbar ist, wird (2) realisiert. Die Leerstelle x.firma wird durch das Translat der PNG von *Lausmann* ausgefüllt. Für die zweite Leerstelle x.ort wird kein Translat gefunden, da es kein entsprechendes Attribut gibt. Deshalb wird dafür eine später zu bindende Variable eingesetzt (in Abb. 8 nicht eingezeichnet). Im nächsten Schritt wird versucht, die zweite Leerstelle, x.int, des PROBE-Pattern auszufüllen mit einem TERM der Sorte "int". Das Translat der DATUMSZAHl 75/, ein TERM der Sorte "int" mit der Form (IN-TEMP 75/), erfüllt die Sortenbedingung und die hier nicht näher genannten syntaktischen Bedingungen für die natürlichsprachliche Struktur. Die letzte Leerstelle des PROBE-Pattern mit der Bezeichnung x.stoffk wird ausgefüllt durch das Translat eines vorangehenden WDET oder aber durch eine mit EXISTVAR gekennzeichnete Variable. Im Beispielsatz existiert ein solches WDET, so daß dessen Übersetzung (LAMDAVAR X2 eingesetzt wird.

Da jedoch das Translat von *Probe* als TERM einsetzbar sein soll, aus dem PROBE-Pattern jedoch eine FORMEL entstanden ist, gibt es praktisch zwei Translate für *Probe*: das ausgefüllte PROBE-Pattern als FORMEL und das letzte Argument der Probe als TERM der Sorte "stoffkoll".

Da die Formel im Verlauf der weiteren Überführung nicht mehr als Argument einer anderen atomaren Formel verwendet werden kann, da Argumente von atomaren Formeln TERME sein müssen, wird eine besondere Struktur auf den vorhandenen Baum "aufgepfropft". Diese Struktur hat die Form



wobei FOR1 ersetzt wird durch die PROBE-Formel, und S die Stelle bezeichnet, an der die vorhandene Gesamtstruktur und später das Translat von S eingesetzt wird.

Als noch zu übersetzendes Symbol der NS-Struktur bleibt *Grenzwerte* übrig. Diesem Symbol wird im Kontext von *überschreiten* das Pattern (GRENZWERTÜBERSCHREITUNG x.dimz x.sit) zugeordnet. Die

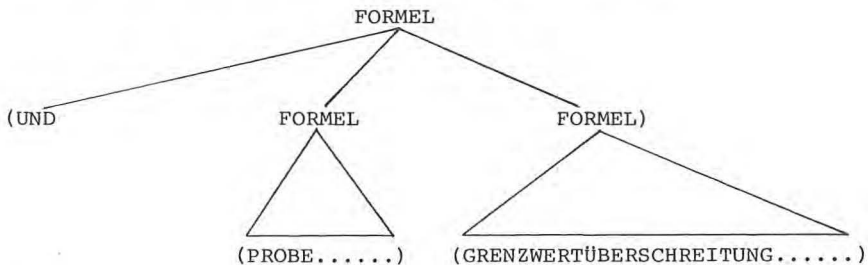
Leerstelle x.dimz erfordert ein Translat der Sorte "dimensionierte Zahl" (dimz). Alternativ dazu ist die Einsetzung eines Patterns (ANTEIL x.stoff x.stoffk x.abstrobj) möglich. Da die erste Alternative nicht realisierbar ist, wird das ANTEIL-Pattern ausgefüllt. Da es für die erste und die letzte Leerstelle keine Einsetzungsmöglichkeiten gibt, werden Variablen eingesetzt (in Abb. 8 nicht eingezeichnet).

Für die Leerstelle x.stoffk wird das Translat einer NG im Nominativ gesucht, die in der Relation ERGÄNZUNG zu *überschreiten* steht. Das Translat muß ein TERM der Sorte "stoffkoll" sein. Diese Bedingungen werden erfüllt vom Translat von *Probe*.

Die Leerstelle x.sit schließlich des Patterns von GRENZWERTÜBERSCHREITUNG wird ersetzt durch eine mit EXISTVAR gekennzeichnete Variable, falls *Grenzwert* nicht durch ein WDET *welch* näher bestimmt ist, was im Beispielsatz jedoch nicht der Fall ist.

Da die so als Translat von *Grenzwert* entstandene FORMEL alle Teiltranslate des Satzes enthält, sofern solche nicht "aufgefropft" wurden, wird diese FORMEL durch Translathebensregeln dem S-Knoten als Translat zugeordnet.

Durch eine Terminierungsregel wird S durch die FORMEL ersetzt, so daß nur noch die folgende Struktur vorhanden ist:



Über dieser Struktur operieren noch zwei Typen von Regeln: der eine Regeltyp legt die Bindung der Variablen entsprechend ihrer Kennzeichnung (LAMBDAVAR, EXISTVAR, FÜRALLVAR) fest; der andere Regeltyp legt fest, wie die Kategoriensymbole FORMEL, TERM etc. durch die terminalen Ketten ersetzt werden, so daß schließlich die Symbolkette

```

(LAMBDA X2
  (EXIST X1.SIT
    (EXIST X3.ORT
      (EXIST X4.STOFF
        (EXIST X5.AOB
          (UND (PROBE (BETRIEB G-L X3.ORT) (IN-TEMP 75/) X2)
            (GRENZWERTUEBERSCHREITUNG
              (ANTEIL X4.STOFF X2 X5.AOB) X1.SIT)))))))

```

entsteht.

5.4. Der Beantwortungsprozeß

Die KS-Frage wird zunächst dem Termitterpreter (TI) übergeben, wo sie als eine Einsetzungsfrage erkannt wird. Der LAMBDA-Term bezeichnet die Elemente der Menge aller im Jahr 1975 bei der Firma Lauxmann gezogenen Proben, die irgendeinen Grenzwert überschritten haben.

Durch "Normalisierung" wird der von der Überführungskomponente gelieferte Ausdruck in den folgenden Ausdruck transformiert:

```

(LAMBDA X2
  (UND (PROBE (EIN (LAMBDA X6 (BETRIEB G-L (EIN (ORT)) X6)))
    (IN-TEMP 75/)
    X2)
    (GRENZWERTUEBERSCHREITUNG (EIN (LAMBDA X7
      (ANTEIL (EIN (STOFF))
        X2
        (EIN (ABSTROBJ))
        X7)))
      (EIN (SIT)))))

```

Es muß also geprüft werden, für welche Proben der angegebenen Art die mit dem Prädikat GRENZWERTUEBERSCHREITUNG beginnende Formel erfüllt ist.

Der TI fragt zunächst nach, ob die Datenbasis irgendwelche Einträge mit diesem Prädikat enthält. Die negative Antwort bewirkt, daß der TI den Theorembeweiser (TB) anstößt, um durch Deduktion eine

Antwort zu gewinnen. Der TB findet folgendes faktenspezifische Axiom bezüglich GRENZWERTUEBERSCHREITUNG:

```
(FUEERALL X.NUM (FUEERALL X.SIT
  (AEQUIV (GRENZWERTUEBERSCHREITUNG X.NUM X.SIT)
    (GROESSER X.NUM (O-GRENZE X.NUM) X.SIT))))
```

Dieses Axiom besagt, daß eine Grenzwertüberschreitung eines Meßwertes genau dann vorliegt, wenn dieser Wert größer ist als die gesetzlich zugelassene Obere-Grenze (Norm) für diesen Stoff.

Unifizierung mit diesem Axiom und der negierten Frage-Klausel aus der GRENZWERTUEBERSCHREITUNG-Formel liefert dem TI folgende GROESSER-Formel zur Auswertung:

```
(GROESSER
  (RESPEKTIV (ANTEIL (EIN (STOFF))
    (PROBE (BETRIEB G-L (EIN (ORT)))
      (EIN (IN-TEMP 75/)))
    (EIN (ABSTROBJ)))
  (O-GRENZE (RESPEKTIV (ANTEIL (EIN (STOFF))
    (PROBE (BETRIEB G-L
      (EIN (ORT)))
      (EIN (IN-TEMP 75/)))
    (EIN (ABSTROBJ))))))
  (EIN SIT))
```

Das KS-Referenzsymbol RESPEKTIV besagt, daß die Elemente der Mengen, die von identischen, mit RESPEKTIV präfigierten Termen bezeichnet werden, "parallel" zueinander in die Formel eingesetzt werden müssen.

Eine so komplexe Frage kann die Datenbasisverwaltung nicht beantworten. Deshalb zerlegt der TI die Frage in folgende Teilfragen und gewinnt die Antwort durch mengentheoretische Operationen über den Teilantworten:

(1) 'Welche Betriebe sind von G-L bekannt?'

Es stellt sich heraus, daß es nur einen einzigen solchen Betrieb gibt, nämlich den Betrieb der Firma G-L in Schwäbisch-Gmünd.

- (2) 'Welche Proben von dieser Firma wurden 1975/ gezogen?'

Die Antwort ist eine Liste von mit UND verknüpften Formeln, wie:

(PROBE §5 75/1//12/11/ØØ §68)

(PROBE §5 75/Ø1/13/14/ØØ §72)

⋮

Die Konstante §5 bezeichnet den Betrieb G-L in Schwäbisch-Gmünd und §68 und §72 die zu den angegebenen Daten gezogenen Proben.

- (3) 'Welche Stoffe wurden in diesen Proben nachgewiesen und wie hoch waren die Meßwerte?'

Als Ergebnis wird eine Liste von ANTEIL-Formeln ausgegeben, z.B.:

(ANTEIL NI1 §68 §169 21.Ø99-MG/L)

(ANTEIL CD1 §68 §169 1.Ø11-MG/L)

(ANTEIL ZN1 §72 §173 Ø.Ø12-MG/L).

Die Konstanten §68 und §72 beziehen sich auf die Probennummern, die Konstanten §169 und §173 auf die Laborberichte für diese Proben.

Die letzten Stellen dieser Formeln werden in einer Liste zusammengefaßt, die das erste Argument der GROESSER-Operation darstellt.

- (4) 'Welches sind die oberen Grenzwerte für diese Stoffe?'

Das Ergebnis ist eine Liste gleicher Länge wie die obige. Sie besteht aus den jeweils oberen Grenzwerten zu den in (3) ermittelten Anteilen und bildet das zweite Argument der GROESSER-Operation.

Die Operation GROESSER prüft nun für jedes Element der Menge der Meßwerte, ob es numerisch größer als der entsprechende Grenzwert ist.

Das Endergebnis besteht aus den Proben, dessen Meßwerte die obere Grenze überschritten haben. Es wird in Form einer Tabelle ausgegeben.

6. Sprachumfang und Leistungsfähigkeit

Die Abgrenzung des Sprachumfangs und der Leistungsfähigkeit von PLIDIS zum heutigen Stand bezieht sich auf die Charakterisierung derjenigen deutschen Sätze, die vollständig vom System verarbeitet werden können, also bis zur Abspeicherung eines Faktums oder zur Beantwortung einer Frage. Die syntaktische Analyse als Einzelkomplex hat eine wesentlich größere Leistungsfähigkeit, da sie die Mehrzahl der deutschen syntaktischen Gebilde analysieren kann. Die Beschränkung der Menge der verarbeitbaren Sätze beruht auf der Unvollständigkeit des Inventars der Überführungsregeln. Es gibt theoretisch keine syntaktischen Strukturen, für die es mit dem vorhandenen Überführungsapparat keine Verarbeitungsmöglichkeit gäbe.

Eine der Zielsetzungen in der PLIDIS-Konzeption war, daß dem Benutzer keine vorgeschriebenen Satzmuster auferlegt werden sollten; letzteres Vorgehen würde keinen echt natürlichsprachlichen Zugriff bedeuten. Der Benutzer ist frei in seiner Formulierung einer bestimmten Fragestellung. Das System kann Paraphrasen verschiedener Art erkennen, wie folgende Beispiele einiger syntaktischen und semantischen Paraphrasen illustrieren:

aktiv-passiv:

- (1) *Hat Brecht die Firma Laußmann geprüft?*
- (1') *Wurde die Firma Laußmann von Brecht geprüft?*

Wortstellungspermutationen:

- (2) *Wurde in der letzten Probe bei Laußmann Eisen festgestellt?*
- (2') *Wurde bei Laußmann in der letzten Probe Eisen festgestellt?*
- (2'') *Wurde Eisen festgestellt in der letzten Probe bei Laußmann?*

Semantische Umformulierungen:

- (3) *Wo hat Brecht Abwasserprüfungen vorgenommen?*
- (3') *Wo hat Weiser Proben gezogen?*
- (4) *Wurden in den letzten drei Proben bei Joos obere Grenzwerte verletzt?*
- (4') *Wurden in den letzten drei Proben der Firma Joos Grenzwertüberschreitungen festgestellt?*

Das verarbeitbare Vokabular besteht aus allen Wörtern, die zur Abfrage der Sachverhalte in den Ausbaustufen I und II (Fragen nach Meßdaten, Proben und Betriebsdaten) notwendig sind. Dieses Vokabular wurde empirisch nach Rücksprache mit dem Benutzer bestimmt.

Die verarbeitbaren syntaktischen Strukturen können grob folgenderweise charakterisiert werden:

- aktive oder passive Hauptsätze im Indikativ, die Nominalgruppen als Ergänzungen oder als Attribute enthalten
- Relativsätze
- negierte und koordinierte Satzteile

Folgender Fragenkatalog enthält Beispielsätze zur Illustration des Sprachumfangs und der Problemlösungsmöglichkeiten von PLIDIS:

Fragen nach Einzeldaten

Inhaltlich vollspezifizierte Fragen:

Enthielt gemäß dem Laborbereich der Chem.-Lwa die am 29.11.75 bei Lauxmann entnommene Probe 0,07 mg/l Cyanid?

Inhaltlich teilspezifizierte Fragen:

Wieviel Chrom wurde bei Lauxmann am 30.11.75 festgestellt?

Mehrfachfragen:

Welcher Prüfer hat bei welcher Firma am 18.2.76 eine Probe entnommen?

Einfache Fragen über Mengen

Entscheidungsfragen:

Hat Fricke die Firmen Heinz Lange und Lauxmann überprüft?

Ergänzungsfragen mit dem Ziel der Aufzählung:

In welchen Betrieben wurden 1976 keine Proben gezogen?

Fragen nach dem Umfang von Mengen

Zählfragen:

In wievielen Proben des Jahres 1975, die Fricke gezogen hat, wurde Eisen nachgewiesen?

Zählfragen mit einer Vergleichsgröße:

*Welche Proben bei Lauxmann enthielten weniger als
1 mg/l Nickel?*

Fragen über Mengen, deren Elemente zeitlich geordnet werden können:

*Wurde in den letzten drei Proben der Firma Joos Cyanid
festgestellt?*

Fragen, zu deren Beantwortung deduziert wird:

*Welche Grenzwerte wurden in der am 20.4.76 bei der D.M.V.
gezogenen Probe überschritten?*

7. Übertragbarkeit des Systems

Bei der Konzeption von PLIDIS wurde berücksichtigt, daß durch die Ausarbeitung theoretisch fundierter und auf größtmögliche Generalität hin konzipierter Lösungsvorschläge die Übertragbarkeit einmal gefundener Lösungen auf neue, vom ursprünglich gewählten Bereich abweichende Anwendungsgebiete gewährleistet ist.

Die entwickelten Verfahren und Darstellungsformen sind von der Form her so allgemein, daß sie nicht qualitativ verändert werden müssen, wenn der Anwendungsbereich gewechselt oder erweitert wird.

Die Voraussetzungen für diese Übertragbarkeit wurden realisiert durch eine Konzeption, die einerseits eine strenge Trennung zwischen den einzelnen Verarbeitungsschritten und andererseits eine strenge Trennung zwischen den Regeln und dem interpretierenden Apparat vorschrieb. Die Anpassung an neue Bereiche oder sogar die Verarbeitung anderer natürlicher Sprachen als das Deutsche erfordert lediglich die Definition neuer Datentypen, die Formulierung neuer Regeln und die Erweiterung der lexikalischen Datenbasis.

Die Erweiterung des morpho-syntaktischen Lexikons wird von automatischen Generierungsverfahren unterstützt. Diese erzeugen von einem eingegebenen Lexem zusammen mit einer dem "Wahrig" (WAHRIG, G., Deutsches Wörterbuch, Gütersloh 1973) entnommenen Flexions-

klassennummer sämtliche Lexikoneinträge für alle flektierten Formen dieses Lexems. (Für den Fall, daß das Wort nicht im Wörterbuch enthalten ist, sind Hilfen zur Bestimmung der Klasse dieses Wortes vorhanden).

Die Netze für die Grammatik des Deutschen sind nahezu vollständig. Diese Netze können aber nach Bedürfnis ohne allzu großen Aufwand erweitert oder durch Alternativen ersetzt werden, falls eine andere Sprache als das Deutsche verarbeitet werden sollte.

Der Hauptaufwand beim Erschließen eines neuen Anwendungsbereiches liegt wohl in der Analyse des Weltmodells und in der anschließenden Formulierung einer KS-Einzelsprache für den gegebenen Weltausschnitt unter Beibehaltung der KS-Syntax und des allgemeinen Teils. Eine solche Analyse ist immerhin für jede neue Anwendung eines Datenbanksystems erforderlich und dürfte im Falle von PLI-DIS keine größeren Schwierigkeiten bereiten als bei sonstigen auf dem Markt angebotenen Systemen.

Anschließend an die KS-Definition müssen die Überführungsregeln für die Symbole der Ausgangssprache definiert werden. Es wäre zu untersuchen, inwieweit computerunterstützte Verfahren zur Definition solcher Regeln möglicherweise eingesetzt werden können.

Der Problemlösungsapparat ist unabhängig vom Inhalt der Datenbank konzipiert. Für den Bereich der klassischen Prädikatenlogik erster Stufe unterliegt das System keinerlei Einschränkungen. Es können alle wohlgeformten KS-Ausdrücke verarbeitet werden. Liegt für den Objektbereich eine Sortenstruktur vor, so muß diese angegeben werden. Das Deduktionsverfahren selbst bleibt von der jeweiligen Sortenstruktur unberührt. Zur Erhöhung der Effizienz des Problemlösungsverfahrens können anwendungsbereichsorientierte Heuristiken angegeben werden. Außerdem müßten zur Lösung komplexer Problemstellungen gewisse faktenspezifische Gesetzmäßigkeiten formuliert werden. Die Formulierung dieser Regeln könnte grundsätzlich auch in natürlicher Sprache erfolgen. Der Problemlösungsvorgang ist nicht abhängig von einem bestimmten Datenbankverwaltungssystem. Es wurde eine Schnittstelle definiert, die minimale Ks-Formeln, die mit jedem herkömmlichen Datenbankverwaltungssystem kompatibel sind, zum Retrieval an die Datenbasisverwaltung übergibt.

Die Übertragbarkeit des Systems auf andere Rechenanlagen wird im nächsten Abschnitt behandelt.

8. Implementierung

Das System PLIDIS ist in INTERLISP, einer Weiterentwicklung der Programmiersprache LISP, geschrieben.

Die Sprache LISP wurde besonders zur Verarbeitung symbolischer Daten entwickelt. INTERLISP (TEITELMAN, 1974) entwickelte sich aus BBN-LISP.

PLIDIS ist implementiert in SIEMENS-INTERLISP (s. EPP, 1977), das die Implementierung der Uppsala-INTERLISP-Version auf einer SIEMENS-4004/151 im BS 2000-Betriebssystem darstellt.

SIEMENS-INTERLISP ist bis auf speicherstrukturabhängige interne Abweichungen kompatibel und auf folgenden Anlagen implementationsfähig:

PDP-10	INTERLISP-10	(Tenex-Timesharing)
IBM-360	INTERLISP-360	(software-paging)
IBM-370	INTERLISP-370	(hardware-paging)
ICL 4/75	INTERLISP-ICL	(hardware-paging)
SIEMENS/4004	INTERLISP-SIEMENS	(hardware-paging)

Unter gewissen Voraussetzungen können die PLIDIS-INTERLISP-Funktionen auch in anderen LISP-Versionen implementiert werden. In diesen Fällen sind diejenigen INTERLISP-Funktionen, die die jeweilige LISP-Version nicht enthält, die aber von PLIDIS benötigt werden, in LISP zu programmieren.

Folgende Anforderungen, die an eine als Muttersprache dienende LISP-Version gestellt werden, sind dagegen von größerer Wichtigkeit, da sie nicht durch zusätzliche Programmierung erfüllt werden können:

Die jeweilige LISP-Version muß die Datentypen

- literale Atome
- Zeichenketten
- Integerzahlen

- Realzahlen

enthalten und außerdem über Funktionen zur Behandlung von Zeichenketten verfügen. Innerhalb von literalen Atomen ist der Punkt als normaler Buchstabe zu behandeln ('A.B' muß also ein literales Atom darstellen). Die Funktionen NCONC, NCONC1 und APPEND müssen auch NIL als erstes Argument zulassen und entsprechend behandeln.

Bei der Implementierung von PLIDIS auf anderen Anlagen als SIE-MENS 4004/151 müssen, außer den oben beschriebenen INTERLISP-Anpassungen, noch einige in ASSEMBLER geschriebene Erweiterungen zur Ein-/Ausgabe beachtet werden. Hierbei sind einige gerätespezifische Tabellen zu ersetzen und die I/O-Macros und eventuell deren Argumente zu modifizieren.

Hinzu kommt noch die Implementierung eines Datenbankzugriffes. Dieser könnte aus einem der kommerziell angebotenen Datenbankverwaltungssysteme bestehen, das an der hierzu vorgesehenen Schnittstelle in PLIDIS angeschlossen werden müßte.

Anmerkungen

- 1 Dieser Abschnitt stellt in komprimierter Form die wichtigsten Aspekte zusammen, wie sie in der Vorhabenbeschreibung zum Vorhaben PLIDIS, 1977 formuliert sind.

PLIDIS ist ein Projekt der Abteilung Linguistische Datenverarbeitung des Instituts für deutsche Sprache. Es wird realisiert in Verbindung mit dem Ministerium für Ernährung, Landwirtschaft und Umwelt des Landes Baden-Württemberg. Die Förderung erfolgt durch den Bundesminister für Forschung und Technologie.

In vielerlei Hinsicht ist PLIDIS als eine Erweiterung des experimentellen Systems ISLIB (Informationssystem auf linguistischer Basis) zu charakterisieren. Innerhalb des Rahmens von ISLIB stand die Erarbeitung theoretischer Grundlagen und die Erprobung verschiedener methodischer und instrumenteller Verfahren im Vordergrund. PLIDIS unterscheidet sich von diesem Vorgängerprojekt insofern, als nun die Betonung darauf gelegt wurde, ein System zu implementieren, das die im Vorgängerprojekt gewonnenen Erfahrungen zu den im Anwendungsbereich "Abwasserüberwachung" konkret vorfindlichen Notwendigkeiten in Beziehung setzt.

- 2 Zu davon verschiedenen Aspekten vgl. Brecht/Lutz, 1977.

- 3 Es können nur eingeschränkte Ausschnitte des Deutschen als Dialogsprache verwendet werden. Diese Ausschnitte genügen in ihrer Vereinfachung nicht den Komplexitätsanforderungen, die sich aus einem Anwendungsbereich für die Ausdrucksweise eines Benutzers ergeben.
Die automatische Übersetzung von anwendungsorientiert formulierten Problemstellungen in geeignete Retrievalabfragen beschränkt sich auf triviale Fälle.
Sofern bestehende Systeme komplexe Retrievalaufgaben bewältigen, ist kein Zugang zur Datenbasis mittels deutscher Sprache möglich.
- 4 Das bedeutet speziell für die Linguistik, daß Sprache im Zusammenhang mit ihrer Verwendung für Problemlösung und Fragebeantwortung betrachtet wird.
- 5 In Abbildung 3 werden die im Sortenbaum angegebenen Sortenbezeichnungen in abgekürzter Form angegeben.
- 6 Über die Realisierung dieser Darstellung in der Datenbasis wird hier nichts ausgesagt; s. hierzu Guntermann in diesem Band.
- 7 Vgl. dazu Lötscher/Kolvenbach in diesem Band.
- 8 Vgl. hierzu Lötscher in diesem Band.
- 9 Vgl. hierzu Wulz in diesem Band.
- 10 Für eine ausführlichere Beschreibung dieser beiden Problemlösungskomponenten s. Dilger und Berry-Rogghe/Dilger in diesem Band.

FORMALE REPRÄSENTATION NATÜRLICHSPRACHLICHER ÄUSSERUNGEN

1. Problemstellung
 - 1.1. Notwendigkeit einer internen Repräsentation
 - 1.2. Funktionen von Repräsentationssprachen in Informationssystemen mit natürlichsprachlicher Ein- und Ausgabe
 - 1.2.1. Semantischsprachliche Funktion
 - 1.2.2. Die IR-Sprache als Modellbeschreibungssprache
 - 1.2.3. Die IR-Sprache als Grundlage des Schlußsystems
 - 1.2.4. Verwendung der IR-Sprache zur Datenbasis-Abfrage
 - 1.3. Typen von Repräsentationssprachen in der KI
 - 1.3.1. Deskriptive vs. prozedurale Semantik
 - 1.3.2. Netzwerkdarstellung und Darstellung durch Formelsprachen
2. Konkretisierung: Die IR-Sprache KS
 - 2.1. Charakterisierung von KS
 - 2.2. Syntaktische Beschreibung von KS
 - 2.2.1. Sorten
 - 2.2.2. Wohlgeformtheitsbedingungen für Formeln
 - 2.2.3. Wohlgeformtheitsbedingungen für Terme
 - 2.2.4. Korrespondenzen zwischen einigen Typen von NS-Nominalphrasen und KS-Termen
 - 2.3. Schlußbemerkung

Anmerkungen

1. Problemstellung und Überblick

1.1. Die Notwendigkeit einer internen Repräsentation

Beim Entwurf von natürlichsprachlichen Informations-, Frage-Antwort- und Problemlösungssystemen wird allgemein davon ausgegangen, daß nicht die natürliche Sprache selbst das Medium der 'internen Repräsentation des Wissens' ist.

Dabei wird unter 'interner Repräsentation des Wissens' die Darstellung aller Informationen verstanden, die als Informationen über den Gegenstandsbereich des Problemlösungs- oder Informationssystems (IS) betrachtet werden im Gegensatz etwa zu Informationen über die Funktionsweise des Systems selbst. Die interne Repräsentation ist dabei diejenige Darstellungsform, über der Fragebeantwortung und Problemlösung durchgeführt werden, ohne daß eine Umformung in eine andere Sprache zwischengeschoben würde.

Das 'Wissen' über den Gegenstandsbereich des Systems kann sich dabei auf sogenanntes 'Weltwissen' über einen konkreten Realitätsausschnitt, eine sogenannte 'Mikrowelt', beziehen, die bei anwendungsbezogenen Systemen gleichzusetzen ist mit dem Anwendungsbeereich; es kann sich jedoch durchaus um jeden möglichen Wissensbereich auch außerhalb einer konkreten Realität handeln. Die Wahl des anthropomorphen Terminus 'Wissen' ist dabei bemerkenswert: Das System teilt mit seinem Konstrukteur und/oder Benutzer dessen Wissen über den Gegenstandsbereich, es übernimmt dessen Sicht, die Welt zu sehen, zu strukturieren und zu interpretieren.

Ausgeschlossen seien bei diesem Begriff des 'Wissens' neben dem Systemwissen auch diejenigen Daten, die sich auf die sprachliche Vermittlung dieses Wissens etwa durch eine natürliche Sprache beziehen. Nicht zum 'Wissen' des Systems im engeren Sinne gehörig ist daher etwa eine syntaktische oder semantische Beschreibung des Deutschen, sofern es sich nicht um ein Informationssystem über Syntax und Semantik des Deutschen handelt, sondern um ein beliebiges Informationssystem mit Deutsch als Interaktionssprache.

Warum nun dient in einem Informationssystem mit beispielsweise Deutsch als Interaktionssprache nicht Deutsch als Sprache der

internen Repräsentation? Normales Deutsch erfüllt aufgrund bestimmter Eigenschaften, die es wohl mit allen natürlichen Sprachen teilt, nicht die Anforderungen, die die formal definierten Problemlösungsoperationen an die ihnen zugrundegelegte interne Darstellung stellen. Solche Eigenschaften sind das Vorhandensein von Mehrdeutigkeiten auf der morphologischen, syntaktischen und lexikalischen Ebene und von Redundanzen ebenfalls auf den verschiedenen Analyseebenen.

Für ein Informationssystem ist beispielsweise die freie Wortstellung, die im Deutschen aufgrund starker morphologischer Kennzeichnung syntaktisch differenter Werte gewährleistet sein kann, leicht verzichtbar, ja vielmehr im Interesse der Ökonomie und Effizienz eher hinderlich. Wird nun also im IS für die Sprachelemente der internen Repräsentation (IR) eine strikte Ordnung festgelegt, so ist die morphologische Diversität überflüssig. Die IR-Sprache kann eine Sprache der unflektierten Stämme sein. Dies gilt analog für syntaktischen und semantischen Variantenreichtum. Es ist ausreichend, wenn es z.B. *e i n e* strukturelle und lexikalische Möglichkeit zur Individuenkennzeichnung gibt gegenüber mehreren wie im deutschen Beispiel:

der Verfasser von Waverley
derjenige, der Waverley verfaßt hat
der Autor von Waverley.

Die Notwendigkeit, daß die Sprache der IR im Verhältnis zur natürlichen Sprache disambiguiert, redundanz- und paraphrasenfrei sei, ist daher unabhängig von der Verarbeitungsqualität allein durch die Bedingungen der Effizienz des Systems gegeben. Bestünde nur sie, so könnte die Sprache der IR ein nach diesen Maßgaben gereinigtes, formalisiertes Deutsch sein.

Soll das System jedoch über die intelligenten Fähigkeiten zur Schlußfolgerung, zur Fragebeantwortung aufgrund von Schlußfolgerungen, zum Entwurf von Handlungsplänen oder zur Rekonstruktion von Geschehnissen aufgrund schlußfolgernder Verknüpfung verfügen, so sind weitergehende Anforderungen an die IR-Sprache gestellt: Über ihr müssen formale Ableitungsregeln definiert sein, d.h. also Regeln, die aufgrund der syntaktischen Form von Einheiten

der IR wirksam werden, etwa nach dem Vorbild logischer Ableitungsregeln wie modus ponens. Sie sind das formale Äquivalent der schlußfolgernden Intelligenz des Systems. Formalisiertes Deutsch kann diesen Anforderungen nicht gerecht werden: Die Definition von formalen Schlußregeln für deutsche Sätze setzt die Explikation ihrer Bedeutung voraus ebenso wie die eindeutige Kenntlichmachung der ermittelten semantischen Funktionen. IR in der Künstlichen Intelligenz (KI) muß daher weitgehend zum Medium der formalen Kunstsprache als IR-Sprache greifen, wenn auch gerade im sprachorientierten Zweig teilweise die Meinung vertreten wird, formalisiertes Englisch oder eine beliebige natürliche Sprache sei in geeigneter Anordnung und mit Rückführung auf Primitive "self explaining".¹

1.2. Funktionen von Repräsentationssprachen in Informationssystemen mit natürlichsprachlicher Ein- und Ausgabe

1.2.1. Semantiksprachliche Funktion

Die formale Kunstsprache der IR steht zu der natürlichen Eingabe- (und Ausgabe-) Sprache (NS) im Verhältnis der 'Semantiksprache'. Eine Sprache L_2 ist Semantiksprache für eine Sprache L_1 , wenn

- (1) mindestens ein Teil aller in L_1 ausdrückbaren Inhalte in L_2 ausdrückbar sind;
- (2) für die beiden Sprachen eine Übersetzungsrelation definiert ist, so daß mindestens ein Teil aller Ausdrücke von L_1 eine Übersetzung in L_2 haben.

Diese Übersetzungsrelation kann bezogen sein auf einen Handlungszusammenhang C .² Dann ist in C ein Ausdruck $a_2 \in L_2$ eine Übersetzung eines Ausdrucks $a_1 \in L_1$ gdw. die Inhalte, die nach der Intention derjenigen, die mit a_1 in C interagieren, mit a_1 in C ausgedrückt werden sollen, auch mit a_2 ausgedrückt werden.

- (3) für die Sprache L_2 eine semantische Interpretation angebar ist, die aufgrund der Übersetzungsrelation auf L_1 induzierbar ist.³

'In C in Übersetzungsrelation stehen' heißt daher nicht interlingual synonym sein. Der Begriff der Synonymie würde voraussetzen, daß alle Inhalte, die beliebige Sprecher von L_1 mit a_1 vermitteln, auch mit a_2 vermittelt werden können. Die beiden Ausdrücke müßten in allen Gebräuchen, die von ihnen regelhaft gemacht werden können, übereinstimmen. Dies ist jedoch im Verhältnis von L_1 und L_2 zumindest immer dann kaum der Fall, wenn L_2 eine formale Kunstsprache ist. Denn L_2 ist in der Regel ärmer als L_1 , da sie ja zu bestimmten Zwecken und für bestimmte Handlungszusammenhänge entworfen ist, etwa zum formalen Schließen o.ä.

Dies gilt in besonderem Maße für Semantiksprachen, die als Repräsentationssprachen in ISen fungieren - etwa im Gegensatz zu rein spracherklärenden Semantiksprachen. Was ein Benutzer des IS mit $a_1 \in L_1$ ausdrücken will, entspricht einem durch den Handlungszusammenhang des IS vorgegebenen eingeschränkten Gebrauch von a_1 , nicht der Totalität der möglichen Gebräuche, die dieser Sprecher von a_1 machen kann.

Aufgrund des zweiten Abschnittes von (2) wird deutlich, daß die Konstruktion von L_2 und die Festlegung der Übersetzungsrelation auf Hypothesen über die Bedeutung von Ausdrücken von L_1 (NS) beruhen, die durch die semantische Interpretation gemäß (3) expliziert werden. Die semantische Interpretation, die für L_2 angegeben wird, ist natürlich nur für diejenigen strukturellen und lexikalischen Ausdrucksmöglichkeiten von NS als indirekte semantische Beschreibung auffaßbar, für die es in L_2 Übersetzungen im Sinne von (2) gibt. Für solche Ausdrücke, die Übersetzungen in L_2 haben, ist die induzierte semantische Beschreibung keine vollständige Interpretation, sondern eine auf den Übersetzungsbegriff bezogene Interpretation.

Die Verwendung einer Semantiksprache in einem IS setzt bestimmte Bedingungen, die die angestrebte linguistisch-semantische Erklärung beeinträchtigen oder zumindest einschränken können. Solche Einschränkungen betreffen z.T. die skizzierten Auswirkungen des Ökonomieprinzips wie Unterdrückung von Paraphrasen, Vernachlässigung bedeutungsleerer und bedeutungsarmer Lexeme in der semantiksprachlichen Übersetzung.

Zum anderen und wichtigeren Teil schränkt die Bindung des Übersetzungsbegriffs und damit der semantischen Interpretation an einen konkreten Handlungszusammenhang die linguistisch-semantische Erklärungsfunktion ein, zumindest, wenn man es als Aufgabe der Semantik ansieht, 'die Bedeutung' eines Ausdrucks vollständig zu beschreiben. Das IS konstituiert den Handlungszusammenhang 'Informationsgewinnung', innerhalb dessen der Mensch-Maschine-Dialog auf das Eingeben und Abfragen faktischen Wissens, gesicherter 'Wahrheiten' ausgerichtet ist. Die Anzahl der innerhalb dieses 'faktischen Diskurses' möglichen Sprachhandlungen ist gegenüber der Diversität natürlichen Sprachverhaltens verschwindend gering. Es werden lediglich Aussagen gemacht und Fragen gestellt. Für Sprachhandlungen wie Zweifeln, Hinterfragen, Lügen oder auch nur 'Nur-Glauben' anstelle von Behaupten ist kein Anlaß gegeben. Allerdings sind solche Informationssysteme des faktischen Diskurses sicherlich nicht die einzige und die einzig erstrebenswerte Ausprägung natürlichsprachlicher Dialogsysteme. Vorstellbar sind daneben zunächst Dialogsysteme zum Zwecke linguistisch-kommunikativer Forschung, in denen unterschiedliche Dialogformen simuliert werden. Inwieweit anwendungsfähige Informationssysteme variierte Formen der sprachlichen Interaktion sinnvoll integrieren können, muß noch überlegt werden. Aber auch in der bisher erprobten pragmatisch so eingeschränkten Form macht die Computersimulation der sprachlichen Interaktion deutlich, in welcher Weise Sprachhandlungen in größere Handlungskomplexe einbezogen sind: Die genuin sprachliche Teilstrategie Informationsgewinnung kann Teil einer übergreifenden Strategie 'Planentwurf', 'Geschehenssteuerung' o.ä. sein.

1.2.2. Die IR-Sprache als Modellbeschreibungssprache

Neben der semantiksprachlichen Funktion, natürlichsprachliche Eingaben in das IS zu interpretieren, sie für die problemlösenden Operationen aufzubereiten bzw. die natürlichsprachliche Ausgabe des Systems vorzubereiten, hat die IR-Sprache die Aufgabe, den Gegenstandsbereich des Systems zu beschreiben, bzw. das 'Wissen'

über die zugrundeliegende Mikrowelt in geeigneter Weise darzustellen.

Beide Aufgaben sind nicht wohl separiert, da ja das Weltmodell über NS-Eingaben und ihre semantiksprachliche Repräsentation aufgebaut werden kann. Das Repräsentationsproblem "Wie wird Weltwissen am geeignetsten strukturiert und durch Zeichen repräsentiert?" tritt jedoch auch in Systemen ohne natürlichsprachlichen Zugang auf, bei denen das Weltwissen intern repräsentiert vor jedem Funktionieren des Systems vorgegeben ist.⁴ Insofern ist diese Funktion von der sprachinterpretierenden systematisch zu trennen.

Interne Repräsentationssprachen in der KI werden meist unter dem Modellbeschreibungsaspekt entworfen und diskutiert - daher auch bei der Namensgebung des Bereichs der direkte Bezug auf das Wissen. Im Vordergrund stehen dabei die Probleme der Konstruktion von Modellen für Mikrowelten, die aus konkreten physikalischen Objekten aufgebaut sind, welche sich in bestimmten dynamisch veränderbaren Beziehungen zueinander befinden. Solche klassischen Mikrowelten der KI sind die Blockwelt WINOGRADS, 1972, die mannigfach variiert wurde oder auch die Situationskonstellation des monkey-banana-Problems.⁵ Neben diesen konkret-dynamischen Modellwelten, die auch um die Problematik der Darstellung von kontinuierlichen Prozessen⁶ erweitert werden können, werden in Informationssystemen außerhalb der KI eher statisch-hierarchische Modelle betrachtet. Sie finden Anwendung, wo es um die Strukturierung eines Wissensbereichs im Sinne einer Klassifikation geht, etwa bei der Thesaurus-Erstellung, bei automatischen Enzyklopädien oder auch bei Datenbanksystemen im organisatorisch-administrativen Bereich.

1.2.3. Die IR-Sprache als Grundlage des Schlußsystems

Es wurde bereits festgehalten, daß die IR-Sprache in einem problemlösenden Informationssystem für formale schlußfolgernde Operationen geeignet sein muß. In der KI wurde das Problem des Repräsentierens zum Zwecke des Schließens völlig abgetrennt von den übrigen Repräsentationsaspekten. Es wurde als das Problem des

automatischen Beweisens von Theoremen⁷ aufgefaßt, die aus einem möglichst abstrakten Gegenstandsbereich wie der Mathematik oder Logik genommen wurden. Hier wurde selbstverständlich auf die formalen Logiksprachen der Aussagenlogik, der Prädikatenlogik erster Stufe und ansatzweise der Prädikatenlogik zweiter Stufe als Repräsentationssprache zurückgegriffen. Diese Ansätze wurden in den 60-er Jahren zu automatischen Beweissystemen des Resolution-Prinzips ausgebaut.

Kam jedoch als zweite Aufgabenstellung neben dem Deduzieren die Modellbeschreibung hinzu, so war der Rückgriff auf Logiksprachen nicht mehr so selbstverständlich. Einerseits wurde versucht, die Beschreibung dynamischer Modelle in einen modalen Logikkalkül bzw. Situationenkalkül⁸ zu integrieren, andererseits wurden - etwa seit 1970 - sogenannte "very high level" Programmiersprachen⁹ entwickelt, mit denen versucht wird, das Problem des Schließens auf eine kalkülunabhängige Weise zu lösen, indem das in einer entsprechenden Modellbeschreibungssprache formulierte Wissen zur Steuerung des schlußfolgernden Programms, zur Selektion und Ordnung plausibler Inferenzschritte herangezogen wird. Diese Vorgehensweise ist in der Literatur mit den Stichwörtern 'prozedurale Darstellung', 'prozedurale Sprachen' oder 'prozedurale Semantik' verbunden.

1.2.4. Verwendung der IR-Sprache zur Datenbasis-Abfrage

Nach semantiksprachlichen Grundsätzen wird man davon ausgehen, daß einzugebende Informationen, also Aussagen, in der gleichen internen Form repräsentiert werden wie Fragen, da beide Sprechakte auf die gleichen propositionalen Gehalte Bezug nehmen. Die IR-Sprache sollte demgemäß Fragen und Aussagen lediglich durch unterschiedliche Sprechaktindikatoren unterscheiden, während die Darstellung von Propositionen oder Satzkernen in Aussagen sich nicht von der Darstellung von Propositionen, die in Fragen enthalten sind, unterscheidet.

Diese grundsätzliche Identität von - technisch gesprochen - Ab-

speicherungs- und Abfragesprache ist in der Praxis der Konstruktion von Datenbanksystemen außerhalb der KI nicht gewährleistet. So werden etwa nach dem CODDSchen Relationenmodell Informationen in einer einfachen relationalen Sprache, deren Einheiten das sogenannte Relationenmodell konstituieren, abgespeichert, während für die Datenabfrage verschiedene Abfragesprachen wie ALPHA, SQUARE¹⁰ hinzudefiniert sind.

Der Versuch, die vier Funktionen einer IR-Sprache in Informationssystemen gegeneinander abzugrenzen, hat wohl deutlich gemacht, daß die linguistisch orientierte semantiksprachliche Funktion durchaus nicht im Vordergrund stehen muß. In der KI spielte und spielt die Modell-Beschreibung und -Handhabung die führende Rolle. Sprachinterpretation ist, z.B. bei WINOGRAD dieser Rolle untergeordnet und stark in sie eingepaßt. Von daher ist auch die bei WINOGRAD, 1972 programmatisch formulierte Benutzung des Weltwissens auf allen Stufen der Sprachanalyse und die nur lasche Unterscheidung von Wissen über die Bedeutung der Wörter und Wissen über das Funktionieren der Blockwelt motiviert.¹¹

1.3. Typen von Repräsentationssprachen in der KI

1.3.1. Deskriptive vs. prozedurale Semantik

In der KI werden gewöhnlich zwei Typen von Repräsentationssprachen unterschieden: deskriptive und prozedurale Sprachen. Deskriptive Sprachen beschreiben die zugrundeliegende Mikrowelt in Form von deklarativen Sätzen; prozedurale Sprachen geben in Form von Anweisungen an, wie die zugrundeliegende Mikrowelt sich zu verhalten hat, bzw. wie mit ihr umzugehen ist. So kann das Wissen über die Situation des Affen-Bananen-Problems - stark vereinfacht - deklarativ wie folgt dargestellt werden:

- (A) Situationsspezifisches Wissen: Ein Affe befindet sich in einem Raum. In einiger Höhe hängt/liegt eine Banane. In dem Raum befindet sich eine Kiste. Die Kiste hat eine solche Höhe, daß der Affe auf ihr stehend die Banane erreichen kann.

- (B) Allgemeinwissen: Affen können sich bewegen. Affen können bewegliche Gegenstände bis zu einem gewissen Gewicht befördern. Affen können klettern usw.
- (C) Folgerungen aus (A) und (B): Der Affe kann die Kiste unter die Banane tragen. Der Affe kann auf die Kiste steigen. Der Affe kann die Banane von der Kiste aus erreichen.

Die entsprechende prozedurale Darstellung setzt das Wissen über mögliche Handlungskonstellationen in Anweisungen zu zielgerichtetem Handeln um:

Du willst die Banane erreichen, die da oben hängt.
 Kannst du dich bewegen, speziell auch klettern?
 Wenn ja, kannst du bewegliche Gegenstände befördern?
 Wenn ja, stelle fest, ob sich in dem Raum, in dem du bist, ein beweglicher Gegenstand befindet?
 Du hast eine Kiste als einen solchen Gegenstand ausgemacht.
 Trage sie unter die Banane!
 Klettere auf die Kiste!
 Strecke deinen Arm aus!
 Kannst du die Banane erreichen?
 Wenn ja, okay.

Übersetzt in das Vokabular einer prozeduralen Sprache entsprechen diesen umgangssprachlichen Handlungsanweisungen sogenannte "Theoreme",¹² die jeweils eine Zielsituation mit den sie sukzessive ermöglichenden Vorgängersituationen verknüpfen. Ein Beispiel für ein Theorem, das einen Ausschnitt des monkey-banana-Problems etwa in Anlehnung an die PLANNER-Theorem-Formulierung WINOGRADS beschreibt, wäre:

```
(DEFTHEOREM TC-GRASP
  (THCONSE (X Y Z) (# GRASP $?X)
    (THGOAL (# HANGING $?X $?Y))
    (THGOAL (# MOVE $?Z (UNDER $?X)) (THUSE TC-MOVE))
    (THGOAL (# CLIMB $?Z) (THUSE TC-CLIMB))
    (THGOAL (# MOVEHAND $?X) (THUSE TC-MOVEHAND))
    (THASSERT (# GRASPING $?X))))
```

Dieses Theorem legt fest, daß ein Gegenstand X ergriffen ist ($\# \text{ GRASP } \#?X$), wenn folgende Teilziele erfüllt sind: Es muß verifiziert sein, daß X an einer Stelle Y hängt ($\# \text{ HANGING } \#?X \#?Y$). Das nächste Teilziel ist, daß ein Gegenstand Z an eine Stelle unterhalb von X bewegt wird; dabei gilt für die Verwirklichung des Teilziels ($\text{THGOAL } (\# \text{ MOVE } \#?Z \text{ (UNDER } \#?X))$) wiederum, daß gewisse Bedingungen erfüllt sein bzw. bestimmte Aktionen ausgeführt werden müssen. Die Verifikation dieser Bedingungen bzw. die Durchführung dieser Aktionen wird durch den Aufruf des Theorems (TC-MOVE) initiiert. Die gleiche Struktur gilt für die folgenden Teilziele 'auf Z steigen' ($\# \text{ CLIMB } \#?Z$) und 'die Hand zu X bewegen' ($\# \text{ MOVEHAND } \#?X$). Sind alle Teilziele erfüllt, so kann die Zielsituation 'X ergreifen' behauptet werden: ($\text{THASSERT } (\# \text{ GRASPING } \#?X)$). Solche Theoreme sind - verglichen mit Logiksprachen - nicht-logische Axiome über Fakten der Mikrowelt, die jedoch nicht aufgrund ihrer syntaktischen Struktur deduktiv verwendet werden, sondern Programme konstituieren, die durch den Interpreter der very-high-level-Programmiersprache interpretiert werden.

Mit der prozeduralen Darstellung handelt man sich auch Probleme ein. So wirkt die prozedurale Darstellung von "Fakten", d.h. wahren Aussagen, gezwungen. Sie müssen als Anweisung, den Satzgehalt als wahr zu betrachten, interpretiert werden.

Am angemessensten erscheint prozedurale Interpretation im Zusammenhang des Nachdenkens über oder Nachspielens von Handlungen oder Prozessen, also der klassischen KI-Konstellation von Ausgangssituation, Zielsituation und der Möglichkeit zielgerichteter Aktionen bzw. der Möglichkeit zur zielgerichteten Initiation von Prozessen. Während in einer deskriptiven Darstellung, wie bereits angedeutet, die Objekte und Relationen der Problemstellung im Prädikatenkalkül oder Situationenkalkül beschrieben werden, wobei nach den üblichen Schlußregeln deduziert wird, und im günstigsten Fall die Beschreibung der Zielsituation nach endlich vielen Deduktionsschritten mithilfe zusätzlicher Axiome aus der Beschreibung der Ausgangssituation ableitbar ist, werden bei prozeduraler Behandlung durch die Anwendung bestimmter "Prozeduren", "Theoreme", "Methoden", "Aktors" - oder wie diese Art von Programmen heißen mögen - sukzessive Transformationen des ursprünglichen Modells

herbeigeführt, die im günstigsten Fall in der Herbeiführung des Zielzustandes enden.

Transformationen sind schlußfolgernde Operationen in einem dynamischen Modell; sie entsprechen daher Ableitungen in einem Situationenkalkül. Das Prinzip ist auch auf nicht-dynamische Modelle übertragbar und somit auf Deduktionen im Sinne der klassischen Prädikatenlogik.

Der Schluß

Alle Menschen sind fehlbar.

Turing ist ein Mensch.

Also ist Turing fehlbar.

konstituiert ebenfalls eine Handlungsanweisung:

Wenn du verifizieren willst, daß Turing fehlbar

ist, so verifiziere, daß er ein Mensch ist.

und somit, ersetzt man 'Turing' durch eine entsprechende Variable, ein PLANNER-Theorem:

(THCONSE (X) (FALLIBLE \$?X)
(THGOAL (HUMAN \$?X))).

Hier wird besonders deutlich, daß das prozedurale Theorem Schlußregeln - unter anderem modus ponens - und das nicht-logische Axiom über den Zusammenhang zwischen Menschsein und Fehlbarkeit integriert.

Auf diese Weise sind Transformations- oder Inferenzschritte, die Ableitungen in der deskriptiven Darstellung ersetzen, besonders eng an das Weltwissen geknüpft. Daraus resultiert, daß das Inferenzverfahren selektiv ist: "World knowledge" steuert die problemlösenden Operationen. Darin sehen ihre Verfechter den Hauptvorteil prozeduraler Sprachen, denn nur so könne wirksam der "kombinatorischen Explosion" rein syntaktischer Verfahren begegnet werden.

Das wesentliche Problem, das durch das prozedurale Konzept aufgeworfen wird, ist das seiner semantischen Explikation und damit zusammenhängend des Nachweises, ob und inwiefern prozedurale Darstellung bestimmten nicht-prozeduralen Repräsentationsformen

vergleichbar oder überlegen ist. Hier scheinen die Meinungen der Verfechter beider Vorgehensweisen dahinzugehen, daß kein grundsätzlicher Deskriptivitätsunterschied vorliegt. Dies gilt dann, wenn man, wie HAYES¹³ vorschlägt, die modelltheoretische semantische Interpretation von Sprachausschnitten in beiden Formalismen einem Vergleich zugrundelegt.

Allerdings ist die Zuordnung von syntaktischer Form und Bedeutung durch eine Interpretationsfunktion für prozedurale Formalismen keineswegs noch so geklärt wie für deskriptive, auf logischer Notation beruhende. Vielmehr wird diese Zuordnung oft erst aufgrund von Übersetzungen in logiksprachliche Ausdrücke rekonstruierbar.

Allerdings war mit der Idee der prozeduralen Repräsentation nicht nur der Wunsch nach inferentieller oder prozeßhafter Modellbeschreibung verbunden, sondern es sollten gleichzeitig auch Strategien für die Funktionsweise des auf die prozedurale Sprache anzuwendenden Interpreters ausdrückbar sein.

HAYES klärt dieses doppelte und in prozeduralen Sprachen nicht immer geglückt realisierte Anliegen als das Bestreben zwei verschiedene "domains" zu repräsentieren, den eigentlichen "domain of discourse", die Mikrowelt, und den Bereich des Systemverhaltens. Beide Bereiche können selbstverständlich in derselben Sprache beschrieben werden, auch und vor allem in Sprachen auf der notationellen Basis der Prädikatenlogik.¹⁴

Die Prozedur, somit eine spezielle Art des Computerprogramms, kommt der Neigung der Computerwissenschaft, Wissen in Berechnungsvorschriften, in Algorithmen einzubetten, entgegen. Darüberhinaus glaubt der mentalistisch orientierte Zweig der KI, im Begriff der Prozedur einen Schlüsselbegriff für die erstrebte Angleichung intelligenter Systeme an die Denk- und Verstehens-"prozesse" des menschlichen Gehirns zu finden. Die kontroversen Ansätze deskriptiver und prozeduraler Repräsentation stehen jedoch auch, wie WINOGRAD feststellt, für die "artificial intelligence incarnation of the old philosophical distinction between 'knowing that' and 'knowing how'".¹⁵ 'Knowing how' bezogen auf semantische Analyse erinnert an WITTGENSTEINS Auffassung von der Bedeutung als Re-

gel des Gebrauchs, ebenso wie die Blockwelt WINOGRADS den Gedanken an manches der Sprachspiele WITTGENSTEINS nahelegt.

Es kann allerdings nicht angenommen werden, daß mit der globalen Anwendung des Prinzips der prozeduralen Interpretation die Probleme 'was ist die Bedeutung dieses Wortes?' oder 'was sind Bedeutungen?' gelöst werden, ebensowenig wie diese Vorgehensweise der differenzierten und eindringenden Analyse WITTGENSTEINS gerecht wird.¹⁶

1.3.2. Netzwerkdarstellung und Darstellung durch Formelsprachen

Wohl eine Ebene unterhalb der Unterscheidung von deskriptivem und prozeduralem Ansatz ist eine weitere Distinktion einzuordnen: die zwischen Netzwerkdarstellung und Darstellung durch Formelsprachen. Zunächst handelt es sich hier um einen Notationsunterschied. Die Netzwerkdarstellung beruht auf graphentheoretischen Begriffen: Ein semantisches Netz ist ein gerichteter Graph, d.h. eine Menge von Knoten und Kanten, wobei eine Kante jeweils zwei Knoten miteinander verbindet. Ein Netz ist dann, wie das folgende einfache Beispiel SCHUBERTS¹⁷ zeigt, nur eine andere - zweidimensionale - Darstellung für eine Menge von geordneten Paaren:



Berücksichtigen wir, daß 'O' den "propositional node" (pn) repräsentiert, so schreiben wir zunächst in geordneter Paar-Schreibweise:

(pn, Edmonton)

(pn, city)

Da die Kanten etikettiert sind, erschöpft die geordnete Paar-Schreibweise jedoch nicht den Informationsgehalt des Netzes. Nehmen wir das Etikett als Relationsnamen für die jeweils zwischen den Elementen des Pairs geltende Relation, so ergibt sich in gewöhnlicher prädikatenlogischer Syntax:

A(pn, Edmonton)

PRED(pn, city).

Diese Rückführung auf prädikatenlogische Syntax zeigt jedoch, daß die konkrete Einsetzung von Knoten- (= Argument-) und Relations- (= Kanten-) Namen und damit die intendierte semantische Interpretation nicht im Einklang ist mit den Konventionen eines üblichen interpretierten Kalküls erster Stufe. Diejenigen Zeichen, die - der gewöhnlichen Interpretation nach - die Eigenschaften von Objekten der zu beschreibenden Welt oder Relationen zwischen ihnen bezeichnen, tauchen nicht als Relationsnamen auf, sondern (hier: city) als Argumentnamen, während die Relationsnamen - in diesem Beispiel - arbiträre Zeichen sind.

Diese Eigentümlichkeit geht zurück auf den zentralen Begriff des Konzept-Knotens, in dem bereits die Unterschiede zwischen Individuum und Menge bzw. Prädikat virtuell aufgehoben sind, da es sowohl Individuenkonzepte als auch Prädikats- und Mengenkonzepte gibt, so daß jeweils zusätzliche Beziehungen zwischen z.B. Individuenkonzepten (Edmonton) und Prädikatkonzepten (city) geschaffen werden müssen, will man sie durch eine etikettierte Kante verbinden.

Diese Technik bietet einer rein formalen semantischen Interpretation möglicherweise wenig Schwierigkeiten, wenn man von den Konzepten als den Individuen des Modells ausgeht. In der Verwendung als Semantiksprache kann man jedoch auf Schwierigkeiten stoßen, da viele semantische Funktionen gerade auf der Unterscheidung von Individuen- und Prädikatsausdrücken und ähnlichen Typunterscheidungen beruhen. Die Differenzierung zwischen Allgemeinbegriffen ("universal concepts") wie 'Stuhl' und den aus ihnen bildbaren Individuenbegriffen wie 'der bestimmte Stuhl 1' kommt dann noch durch die unterschiedliche Benennung von Knoten mit einerseits 'Stuhl' andererseits 'Stuhl 1', 'Stuhl 2' usw. wieder herein, ohne daß die Zusammenhänge zwischen Allgemeinbegriff und zugehörigem Individuenbegriff, die in der natürlichen Sprache syntaktisch wohldeterminiert und durch entsprechende semantische Regeln interpretiert sind, in dieser Notation ersichtlich würden.

Auch die Etikettierung der Kanten mit bedeutungsleeren labels wie A, B und C in den Beispielen SCHUBERTs wirkt bei der sprachinterpretierenden Anwendung von Netzwerksprachen störend. Ihre Erset-

zung durch potentiell bedeutungsvolle Etiketten wie Agens, Rezipiens, Objekt führt zu einer tiefenkasus-orientierten Variante des semantischen Netzes, die durch SCHANK und seine Mitarbeiter, BRUCE, SIMMONS u.a. im Bereich der KI vertreten ist. Die auf FILLMORE zurückgehende Repräsentation des einfachen Satzes auf der Basis von Kasusrahmen der Prädikatsverben wird in der KI durch die Auffassung von Kasusrahmen-Strukturen als konzeptuelle Strukturen in höherem Maße als in der entsprechenden linguistischen Literatur mentalistisch gefärbt. Dabei macht SCHANK mit der Setzung von nur wenigen primitiven Akten oder Aktionen, um die entsprechend Kasusrahmen Aktanten arrangiert sind, recht starke Hypothesen über für alle Sprachteilhaber gleichartige "Konzeptualisierungen" von Handlungen und Sprachhandlungen, denn durch diese wenigen "primitive acts" soll ein wesentlicher Teil menschlicher Weltfassung abgedeckt sein.

SCHUBERT unternimmt neben anderen¹⁸ den Versuch, die primitive Form des semantischen Netzes, in dem mehrere Konzeptknoten durch etikettierte Kanten verbunden sind, so zu erweitern, daß die Ausdrucksmöglichkeiten der Prädikatenlogik erster Stufe erreicht werden. Zu diesem Zweck müssen Notationsweisen für Junktoren, Quantoren und Variablen eingeführt werden. Dieser Versuch zeigt noch deutlicher als die Repräsentation im einfachen Netz, daß es lediglich um die graphische Umsetzung bereits festgelegter syntaktischer Ausdrucksmöglichkeiten geht.

So erörtert SCHUBERT z.B. die graphische Verdeutlichung von Bezüglichkeiten, für die es in der linearen Schreibweise feste Konventionen gibt, wie die Anwendung von Junktoren auf die Gesamtproposition - sprich den Propositionalknoten - und die Repräsentierbarkeit von quantifizierten Variablen durch spezielle Knotenarten.¹⁹ Dabei bedarf im besonderen die Umsetzung der in der prädikatenlogischen Schreibweise durch die lineare Ordnung ausgedrückten Skopusverhältnisse in die zweidimensionale Darstellung besonderer Überlegung.

Der Vorteil von Netzwerksprachen, die, wie gezeigt wurde, im wesentlichen als notationelle Varianten bestehender Repräsentationsformen angesehen werden können, soll darin liegen, daß durch die

intendierte - allerdings nicht immer eingehaltene - eindeutige Entsprechung zwischen nur einmal repräsentiertem Knoten und Konzept, also zwischen Zeichen und Bedeutung bzw. Repräsentationseinheiten und Wissenskomplexen, ein strukturiertes Modell "analogischen" Charakters²⁰ entsteht. Diese analogische Organisation entspricht den Vorstellungen über optimale Datenstrukturen, wo alle Informationen zu einem "entry" zu einer "unique computer storage location"²¹ gruppiert werden.

Daher werden Netzwerkdarstellungen als für die Zwecke des information retrieval besonders geeignet erachtet: Es ist jeweils nur notwendig, den Kanten zu folgen, die Knoten mit ihren Nachbarknoten zu verbinden, um relevante konzeptbezogene Informationen aufzufinden. Die netzwerkmanipulierenden Prozeduren haben genau diese Funktion.

Dieser Strukturgesichtspunkt kann jedoch auch statt im technischen in einem mentalistischen Sinne interpretiert werden, ähnlich wie dies beim prozeduralen Ansatz der Fall ist. Dann werden semantische Netze als memory-Modelle bezeichnet und als Analoga des menschlichen Gedächtnisses aufgefaßt.²²

Wie eingangs angemerkt, liegt die Differenzierung Netzwerksprache versus Formelsprache eine Ebene unterhalb der Distinktion deskriptiv - prozedural. Prinzipiell ist daher letztere Differenzierung auf Netzwerkrepräsentationen anwendbar: Werden Netze prozedural interpretiert, so werden mit den primitiven Knoten des Netzes jeweils Inferenz-Prozeduren assoziiert. Wird im Sinne deklarativer Repräsentation auch Regelwissen im Netz dargestellt, so kommt dem Netzwerkinterpret die Funktion eines Deduktionsmechanismus nach Theorembeweiser-Art zu. Die Anpassung von für die Syntax von Formelsprachen definierten Schlußregeln an das Zeichensystem von Netzwerksprachen - bzw. ihre notwendige Linearisierung in den Datenstrukturen des Computersystems - beruht allerdings auf einer vollständigen Reduzierbarkeit der Netzwerksprache auf den Prädikatenkalkül erster Stufe, was seinerseits die Wahl der speziellen Notationsweise fragwürdig macht.

2. Konkretisierung: Die IR-Sprache KS

2.1. Charakterisierung von KS

Die interne Repräsentationssprache, wie sie im Rahmen der Arbeiten an PLIDIS entwickelt wurde, die sogenannte 'Konstruktssprache' (KS) ist eine deskriptive Formelsprache, die sich am Prädikatenkalkül erster Stufe (PK) orientiert und gewisse Erweiterungen gegenüber PK aufweist. Sie hat in einem IS, speziell PLIDIS, genau die beschriebenen Funktionen der semantiksprachlichen Interpretation eines Ausschnitts einer natürlichen Sprache und der Modellbeschreibung; sie dient als Abspeicherungs- und Abfragesprache und ist sprachliche Basis des Problemlösers.

Primär waren jedoch die Gesichtspunkte der Interpretation natürlichsprachlicher Äußerungen und des Geeignetseins zum formalen Problemlösen. Die zu konstruierende IR-Sprache sollte daher der natürlichen Sprache, konkret dem Deutschen, möglichst "nahe" sein und möglichst nicht so stark vom PK abweichen, daß Theorembeweise nach dem Vorbild des PK unmöglich würden. Der Begriff der Nähe zu NS ist dabei so zu verstehen, daß

- a) der Übersetzungsbegriff im oben beschriebenen Sinne für das Verhältnis eines Ausschnitts von NS zu KS erfüllt ist;
- b) eine weitgehende Übereinstimmung in den Strukturprinzipien von NS und KS - im betrachteten Ausschnitt - vorliegt.

Anforderung b) ist sowohl von praktischem Interesse für die Definition einer effizienten Übersetzungsgrammatik als auch von theoretischer Bedeutung, wenn KS als Semantiksprache für NS, bzw. einen Ausschnitt einer NS fungieren soll.

Dabei setzt die grundsätzliche Entscheidung für eine Orientierung an PK einer Strukturentsprechung im Sinne von Oberflächennähe in folgenden Punkten Grenzen: Wortartunterscheidungen zwischen Adjektiven, Nomina und Verben werden nicht gemacht.²³ Ihnen entsprechen wie in PK in KS Prädikatsausdrücke. Ebenso steht die Beachtung des Ökonomieprinzips der Oberflächennähe von KS entgegen: Semantisch leere oder vermeidbare - d.h. in ihrer semantischen Funktion auf andere Weise erfaßbare - Ausdrücke der natürlichen

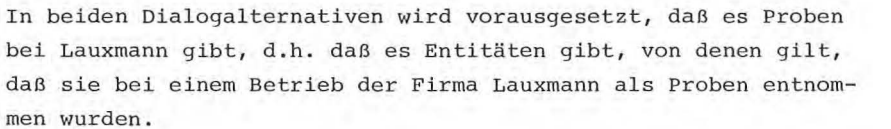
Sprache werden in KS nicht repräsentiert. Es werden daher Kasus- und Verbalendungsmorpheme unterdrückt, ebenso nicht-kommutierbare Präpositionen. KS befindet sich damit im Gegensatz zu rein erklärenden formalen Semantiksprachen, wie etwa den kategorialen Sprachen MONTAGUES oder CRESSWELLS, wo zumindest tendenziell für jeden Ausdruck eines gewählten NS-Ausschnitts eine semantiksprachliche Entsprechung definiert ist. Der Ausschnitt des Deutschen, für den KS als Semantiksprache angesehen werden kann, ist unter zwei völlig unterschiedlichen Gesichtspunkten auszugrenzen, nämlich nach

- dem Umfang des per Übersetzungslexikon als Entsprechung von deutschen Vokabular festzulegenden KS-Vokabulars;
- den strukturellen Ausdrucksmöglichkeiten.

Die Festlegung des KS-Vokabulars ist abhängig von der darzustellenden Mikrowelt und dem, meist fachsprachlichen, Vokabular, mit dem über die Mikrowelt geredet wird. Die bereichsabhängigen Vokabularfestlegungen für KS konstituieren jeweils eine 'KS-Einzelsprache'. Ein Beispiel für eine KS-Einzelsprache ist 'KS-Abwasserüberwachung'. Eine KS-Einzelsprache läßt semantiksprachliche Interpretation für genau den natürlichsprachlichen Ausschnitt zu, mit dem Informationen über den gewählten Weltausschnitt formuliert werden. Dieser im Beispielfall 'Abwasserüberwachung' zum Teil fachsprachliche Sprachausschnitt enthält jedoch immer einen starken Anteil Allgemeinwortschatz, für den in der KS-Einzelsprache ebenfalls Entsprechungen definiert sein müssen.

Die strukturellen Ausdrucksmöglichkeiten der IR-Sprache werden durch diejenige Untermenge von Sprachhandlungstypen, die in einem IS bestimmter Qualität vollziehbar sein müssen, systematisch determiniert. Auf diese Weise ergibt sich eine funktionale Bestimmung für den Grad der Nähe zu NS. Es handelt sich dabei im Falle von PLIDIS-KS um die Sprachhandlungstypen Frage und Antwort in einem faktischen Diskurs (wie in Abschnitt 1.2.1. eingeführt). Innerhalb dieser unabhängigen Sprachhandlungen muß insbesondere die Sprechhandlung 'Referenz' in vollem Umfang der natürlichsprachlichen Möglichkeiten in der Interaktion mit dem IS vollziehbar sein. Denn in der Kommunikation mit einem IS ist die Nennung

Eng damit verknüpft ist die Notwendigkeit, in IR die Existenzpräsuppositionen des natürlichsprachlichen faktischen Diskurses zu erhalten. Solche Präsuppositionen sind nämlich an die Sprechakte der Referenz mithilfe von definiten Nominalphrasen in Frage-Antwort-Dialogen geknüpft. Als Beispiel diene die folgende alternative Entwicklung eines simplen Dialogs aus dem PLIDIS-Anwendungsbereich:



112

Es gibt ja keine solchen Proben.

bzw.

Über die Existenz solcher Proben ist dem IS ja nichts bekannt.

Nun ist es mit den Mitteln des klassischen Prädikatenkalküls nicht möglich, Existenzpräsuppositionen auszudrücken: Ausdrücke, mit denen Sprecher von NS solche Präsuppositionen verbinden – definite Nominalphrasen –, werden im reinen Prädikatenkalkül auf Existenz *a u s s a g e n* zurückgeführt. Eine solche Rückführung ist kommunikativ unangemessen: Sie würde z.B. im obigen Dialog zur Antwort *nein* führen, wenn es keine Proben bei Lauxmann gibt.

Die Konstruktsprache weicht daher in der Behandlung referentieller Ausdrücke von PK ab. Sie versucht den natürlichsprachlichen Möglichkeiten durch einen ausgebauten Termbildungsmechanismus nachzukommen. In ihm soll die Bildung von Übersetzungsäquivalenten für definite und indefinite NP im Singular und Plural ermöglicht werden. Ebenfalls müssen Entsprechungen für kennzeichnende Ausdrücke, die natürlichsprachliche Relativsätze oder attributive Adjektive enthalten, definierbar sein.

Das Prinzip bei der Festlegung und Differenzierung von Termen ist selbstverständlich jedoch nicht an der NS-Ausdrucksseite orientiert, sondern an ihrer logisch-semantischen Qualität: KS differenziert zwischen Individuentermen, Bezeichnungen für Elemente von Mengen, Bezeichnungen für Mengen. Auf der natürlichsprachlichen Seite fungieren singularische Nominalphrasen als Bezeichnungen für Individuen, wobei allerdings dieser ausdrucksseitigen Kategorie auch andere Funktionen (generischer Gebrauch des definiten Artikels usw.) zukommen. Pluralische Nominalphrasen bezeichnen gewöhnlich Elemente von Mengen, nicht Mengen im logischen Sinn einer gegenüber Individuenausdrücken höhertypisierten Einheit. Nur in wenigen Kontexten, z.B. in Kookkurrenz mit Ausdrücken wie *in der Mehrheit sein*, *in der Minderheit sein*, *weniger werden* bezeichnen pluralische NP echte Mengen. Beispiele sind:

Die Männer waren gegenüber den Frauen in der Minderheit.

Schornsteinfeger werden immer weniger.

In der folgenden Beschreibung wird die Behandlung der höherstufigen Entität Menge ausgeklammert. Dagegen werden Bezeichnungen für die Elemente einer Menge als nicht-höherstufige Zusammenfassungen von Individuen, die wir 'Listen' nennen, behandelt. Den Ausdruck *die Kinder* im folgenden Beispielsatz fassen wir als Bezeichnung für eine solche Liste auf, die in diesem Falle die Zusammenfassung derjenigen Individuen ist, von denen gilt, daß sie Kinder von Hans sind:

Die Kinder von Hans spielten im Garten.

Dieser Satz macht keine Aussage über die Menge 'die Kinder von Hans', sondern über die Elemente dieser Menge.

Das semantiksprachliche Interesse an der adäquaten Repräsentation von referierenden Phrasen trifft sich mit der Notwendigkeit, bei der Datenbasisabfrage auf Mengen von Objekten zuzugreifen und mengentheoretische Operationen zu initiieren: Im Falle einer natürlichsprachlichen Abfrage ist ein Ausdruck wie

*Welche Proben wurden bei der Firma Meier und bei
der Firma Kraus im Jahre 1977 entnommen?*

als Aufforderung, die Vereinigungsmenge der Menge der Proben der Firma Meier im Jahre 1977 und der Menge der Proben der Firma Kraus im Jahre 1977 zu bilden, aufzufassen. Die NS-Ausdrücke *welche* und *und* sind als Indikatoren für die auszuführenden Mengenoperationen zu betrachten. In KS dient ein Term, der die Elemente der entsprechenden Menge bezeichnet, als Entsprechung des NS-Ausdrucks. Er wird bei der Abfrage vom Terminierteprer im Sinne der Ermittlung der jeweiligen Menge aus der Datenbasis interpretiert (vgl. BERRY-ROGGHE/DILGER in diesem Band).

2.2. Syntaktische Beschreibung von KS

Die folgende Skizze der Syntax von KS wird ergänzt durch informelle Bedeutungserklärungen für Terme, da diese Ausdrücke aufgrund der Semantik des PK nicht vollständig erklärt sind. Es wird die

Syntax von KS - P r o p o s i t i o n e n beschrieben, pragmatische Operatoren sind nicht aufgeführt.

2.2.1. Sorten

In KS wird das natürlichsprachliche Prinzip der Regelung von Ko-
okurrenzen zwischen Teilausdrücken durch semantische Selektions-
beschränkungen übernommen, und zwar in der auf der logischen Ebene
entsprechenden Form des sortierten Kalküls.

Auf der semantischen Ebene bedeutet das: Ebenso wie Selektionsre-
striktionen der natürlichen Sprache eine aufgrund der Kommunika-
tion in dieser Sprache und für die Kommunikation mit dieser Spra-
che gültige Strukturierung des zugrundeliegenden Weltmodells an-
geben, setzen KS-Aussagen einen strukturierten Individuenbereich
voraus, in dem Individuen in Sorten klassifiziert sind. Die Sor-
ten bilden eine gerichtete Menge bezüglich der Mengeninklusion.

Die Festlegung konkreter Sorten wird im Rahmen von PLIDIS vom je-
weiligen Weltmodell abhängig gemacht und daher im Rahmen von KS-
einzelsprachlichen Festlegungen abgehandelt.

Die Sortenfestlegung für die KS-Einzelsprache 'Abwasserüberwa-
chung' wird im Beitrag von BERRY-ROGGHE et al. in diesem Band an-
gegeben. Damit wird für den gegenwärtigen Stand darauf verzichtet,
- den Zusammenhang zwischen ausgangssprachlichen semantischen Re-
striktionen und bereichsorientierten Klassifizierungen in KS-
Sorten exakt zu klären;
- die Grenzen bzw. Überschneidungen zwischen fachsprachlicher
und gemeinsprachlicher Sortenstruktur zu definieren.

Für die Syntax von KS resultiert aus der Sortiertheit des Indivi-
duenbereichs, daß Individuensymbole sortierte Symbole sein müssen
und für jedes n-stellige Prädikatsymbol das n-Tupel seiner Argu-
mentsorten definiert sein muß. Es ist daher eine Menge von Sorten-
symbolen gegeben, auf der eine Halbordnung (in Zeichen \leq) defi-
niert ist, die als Mengeninklusion zwischen Sorten interpretiert
wird. Da in diesem Zusammenhang auf die Festlegung konkreter Sor-
ten verzichtet wurde, sind die Sortensymbole abstrakter Natur.

Die Menge S der Sortensymbole sei

$$S = \{s_i \mid i \in \mathbb{N}\}.$$

Zu der Funktion, Selektionsbeschränkungen der natürlichen Sprache wiederzugeben, kommt hinzu, daß die Sortiertheit der Sprache Vorteile für die Verarbeitung in einem IS mit sich bringt: Der Überführungsalgorithmus von PLIDIS benutzt die Sorten zur Selektion geeigneter Übersetzungsentsprechungen; der Problemlöser benutzt die Sorten zur Einschränkung des Suchraums, da bei Vorgabe sortierter Terme sowohl beim Termitterpreter als auch beim Theorembeweiser der Individuenbereich, der mögliche Belegungen des gesuchten Terms liefert, auf die Sorte, die dem jeweiligen sortierten Term entspricht, eingeschränkt wird.²⁵

In der vorliegenden Definition von KS werden alle Terme so aufgefaßt, daß ihnen als semantische Interpretationen Elemente des sortierten Individuenbereichs zugeordnet werden. Es wird nicht berücksichtigt, daß es möglicherweise auch für eine IR-Sprache notwendig ist, neben der Extension von Ausdrücken ihre Intension zu berücksichtigen, ebensowenig wie daß es eventuell auch bezeichnende Ausdrücke für Entitäten höheren Typs, also Eigenschaften, "relations-in-intension" usw. geben muß.

2.2.2. Wohlgeformtheitsbedingungen für Formeln

In KS wird eine Unterscheidung zwischen Operationssymbolen und gewöhnlichen Relationssymbolen gemacht, wobei Operationen wie üblich als mehrdeutige, d.h. an der letzten Stelle eindeutige Relationen aufgefaßt werden. Diese Distinktion wird für die Bildung von Formeln nicht relevant, da hier beide Symbolarten nur in ihrer Funktion als Prädikatssymbole verwendet werden. Sie spielt nur bei der Bildung von Termen eine Rolle.

/1/ Operationssymbole und Relationssymbole von KS sind Prädikatssymbole von KS.

/2/ Wenn P ein n -stelliges Prädikatssymbol ist,

$$\langle s_{i_1}, \dots, s_{i_n} \rangle$$

das Argumentsortentupel von P und

$$t_1^{s_{i_1}}, \dots, t_n^{s_{i_n}}$$

Terme jeweils der Sorten s_{i_1}, \dots, s_{i_n} sind, dann ist

$$(P \ t_1^{s_{i_1}} \ \dots \ t_n^{s_{i_n}})$$

eine atomare Formel von KS.

Beispiel:

$P = \text{KOLLEGE Argumentsortentupel: } \langle \text{natper}, \text{natper} \rangle$

Es seien HANS und FRANZ Terme der Sorte natper, dann ist

(KOLLEGE HANS FRANZ)
'Franz ist Kollege von Hans'

eine atomare Formel von KS.

Kommentar:

Zur Wiedergabe von Verbalsätzen in NS, die Aussagen über zeitlich festgelegte Handlungen, Zustände oder Vorgänge machen, dienen in KS Aussagen, die aus Prädikaten, speziell Operationssymbolen mit folgendem Argumentsortentupel gebildet werden:

$\langle s_{i_1}, \dots, s_{i_{n-2}}, \text{int}, \text{sit} \rangle$ bzw.

$\langle s_{i_1}, \dots, s_{i_{n-2}}, \text{int}, \text{akt} \rangle$ bzw.

$\langle s_{i_1}, \dots, s_{i_{n-2}}, \text{int}, \text{zus} \rangle$ usw.

wo das Sortensymbol int die Sorte 'Zeitintervall' bezeichnet und die Sortensymbole akt ('Aktion') und zus ('Zustand') als Untersorten der Sorte 'Situation' (sit) aufzufassen sind.

Die NS-Aussage

Hans kommt am 23.7.77 von Frankfurt nach Mannheim.

wird mit Hilfe von

$P = \text{KOMM Argumentsortentupel: } \langle \text{natper}, \text{ort}, \text{ort}, \text{int}, \text{akt} \rangle$

wie folgt wiedergegeben

(KOMM HANS FRANKFURT MANNHEIM 77/07/23 AKTn)

wo AKTn der Name derjenigen Aktion ist, die darin besteht, daß Hans am 23.7.77 von Frankfurt nach Mannheim gekommen ist. Für solche Aktions- und Situations-Entitäten kennt NS keine Namen, sie werden im IS als Kombinationen von Sortensymbol und Nummern erzeugt und vergeben.

Aufgrund der Tatsache, daß Verbalsätze jeweils mithilfe expliziter Zeitangaben in KS wiedergegeben werden, ist die Berücksichtigung von Tempora, also Zeitstufenbezeichnungen, verzichtbar. Die Einordnung als vergangen oder gegenwärtig ergibt sich für den Benutzer des IS aus dem Verhältnis des im KS-Satz angegebenen Zeitintervalls zu der realen Dialogzeit. Die deiktische Komponente bei Zeitangaben und Verbaltempora wird daher aus KS verlagert, deiktische Zeitadverbien wie *heute*, *gestern* müssen durch die Überführungskomponente auf kalendarische Daten abgebildet werden. Kalendarische Daten werden in den Beispielen vereinfachend als Individuenkonstanten repräsentiert.²⁶

Nicht-atomare Formeln werden mithilfe der gewöhnlichen Junktoren und Quantoren des PK erster Stufe gebildet, wobei wir von der polnischen Präfixnotation Gebrauch machen. Die Menge der Formeln von KS kann daher rekursiv wie folgt definiert werden:

/3/ Atomare Formeln sind Formeln.

/4/ Es seien NICHT (einstellig), UND, ODER, IMPLIK, AEQUIV (zweistellig) die aussagenlogischen Junktoren, p und q seien Formeln von KS, dann sind

(NICHT p)
(UND p q)
(ODER p q)
(IMPLIK p q)
(AEQUIV p q)

Formeln von KS.

/5/ Es seien FÜRALL und EXIST die Quantoren, x^s_i eine Variable der Sorte s_i und p eine Formel, dann sind

(FÜRALL x^s_i p)
(EXIST x^s_i p)

Formeln von KS.

2.2.3. Wohlgeformtheitsbedingungen für Terme

/6/ Sortierte Variable und Konstanten sind Terme von KS.

Beispiel für Individuenkonstanten:

HANS Sorte: natper

SITØ1 Sorte: sit

FRANKFURT Sorte: ort

/7/ Es sei p eine Formel, x^{s_i} eine Variable der Sorte s_i , dann ist

$(\text{LAMBDA } x^{s_i} p)$

ein Term von KS, dessen Sorte s_i ist.

Beispiele:

- (1) $(\text{LAMBDA } x^{\text{natper}} (\text{VATER } x^{\text{natper}} \text{ HANS}))$
'diejenige Person/diejenigen Personen, deren Vater Hans ist' \Leftrightarrow 'die Kinder von Hans'.
- (2) $(\text{LAMBDA } x^{\text{natper}} (\text{KOLLEGE HANS } x^{\text{natper}}))$
'diejenigen Personen, die Kollegen von Hans sind'
 \Leftrightarrow 'die Kollegen von Hans'.
- (3) $(\text{LAMBDA } x^{\text{natper}} (\text{EXIST } y^{\text{natper}} (\text{KOLLEGE } x^{\text{natper}} y^{\text{natper}})))$ 'diejenigen Personen, die Kollegen haben'.
- (4) $(\text{LAMBDA } x^{\text{natper}} (\text{UND } (\text{KOLLEGE HANS } x^{\text{natper}}) (\text{VATER } x^{\text{natper}} \text{ FRANZ})))$
'diejenigen Personen, die sowohl Kollegen von Hans als auch Kinder von Franz sind'.

Informale semantische Erklärung:

Durch einen LAMBDA-Term bezeichnen wir 'Listen' von Individuen, die die Formel p bezüglich der abstrahierten Stelle erfüllen, d.h. die Elemente der Extension des Prädikats

$(\text{LAMBDA } x^{s_i} p)$

Mit (1) bezeichnen wir somit die Elemente der Menge

$\{x \mid (\text{VATER } x \text{ HANS})\}$

/7.1./ Wenn p eine atomare Formel ist, deren letztes Argument abstrahiert ist, d.h. wenn der LAMBDA-Term die Form

$$(\text{LAMBDA } x^{s_{i_n}} (p \ t_1^{s_{i_1}} \ \dots \ t_{n-1}^{s_{i_{n-1}}} \ x^{s_{i_n}}))$$

hat, wo s_{i_1}, \dots, s_{i_n} das Sortentupel von P ist, schreiben wir zum Zweck der Abkürzung

$$(p \ t_1^{s_{i_1}} \ \dots \ t_{n-1}^{s_{i_{n-1}}})$$

Beispiel:

$$(\text{LAMBDA } x^{\text{natper}} (\text{KOLLEGE HANS } x^{\text{natper}}))$$

kann abgekürzt werden als

$$(\text{KOLLEGE HANS}).$$

Als Zeichen für die Gleichheit von Formeln aufgrund dieser Abkürzungskonvention verwenden wir '='.

/7.2./ LAMBDA-Terme, die aus atomaren Formeln gebildet sind, deren Prädikatsymbol bezüglich der abstrahierten Stelle eindeutig ist, sind Individuenterme. Individuenterme bezeichnen das Element einer einelementigen Liste. Alle anderen LAMBDA-Terme sind Listenterme.

/7.3./ Wenn in einem Term

$$\begin{aligned} &(\text{LAMBDA } x^{s_{i_1}} (p \ t_1^{s_{i_1}} \ \dots \ t_{n-1}^{s_{i_{n-1}}} \ x^{s_{i_1}})) \quad = \\ &(p \ t_1^{s_{i_1}} \ \dots \ t_{n-1}^{s_{i_{n-1}}}) \end{aligned}$$

P ein Relationssymbol ist, so liegt bezüglich der abstrahierten Stelle keine Eindeutigkeit vor. Somit ist der Term nach /7.2./ ein Listenterm. Wir nennen Terme dieser Form relationale Listenterme.

Beispiel:

$$\begin{aligned} &(\text{LAMBDA } x^{\text{natper}} (\text{KOLLEGE HANS } x^{\text{natper}})) \quad = \\ &(\text{KOLLEGE HANS}) \\ &\text{'die Kollegen von Hans'}. \end{aligned}$$

/7.4./ Operationssymbole von KS werden nach ihrem Verhalten bei der Termbildung in zwei Gruppen differenziert:

Strikt zuordnende Operationssymbole sind eindeutig bezüglich der letzten Stelle, auch wenn die Argumentterme Listenterme sind.

Zuordnend singuläre Operationszeichen sind eindeutig bezüglich der letzten Stelle nur dann, wenn alle Argumentterme Individuenterme sind.

/7.5./ Nach /7.2./ und /7.4./ sind Terme der Form

$$\begin{aligned} & (\text{LAMBDA } x^{s_i} (F t_1^{s_{i_1}} \dots t_{n-1}^{s_{i_{n-1}}} x^{s_i})) = \\ & (F t_1^{s_{i_1}} \dots t_{n-1}^{s_{i_{n-1}}}) \end{aligned}$$

wo F ein strikt zuordnendes Operationssymbol ist, Individuenterme von KS.

Beispiele:

- (5) $(\text{LAMBDA } x^{\text{akt}} (\text{KOMM HANS FRANKFURT MANNHEIM } 77/07/23 x^{\text{akt}})) =$
 $(\text{KOMM HANS FRANKFURT MANNHEIM } 77/07/23)$
 'das Kommen von Hans von Frankfurt nach Mannheim
 am 23.7.77'
- (6) $(\text{LAMBDA } x^{\text{akt}} (\text{KOMM } (\text{LAMBDA } x^{\text{natper}} (\text{KOLLEGE HANS } x^{\text{natper}})) \text{ FRANKFURT MANNHEIM } 77/07/23 x^{\text{akt}})) =$
 $(\text{KOMM } (\text{KOLLEGE HANS}) \text{ FRANKFURT MANNHEIM } 77/07/23)$
 'das Kommen der Kollegen von Hans von Frankfurt nach
 Mannheim am 23.7.77'.

Kommentar:

Strikt zuordnend sind Operationssymbole mit dem Sortentupel

$$\langle s_{i_1}, \dots, s_{i_{n-2}}, \text{int}, \text{sit} \rangle$$

bzw. mit den Untersorten von sit anstelle von sit selbst, also die KS-Entsprechungen von NS-Verben. Wir nehmen an, daß solche KS-Prädikate bezüglich der letzten Stelle, also bezüglich der bezeichneten Situation eindeutig sind, auch wenn die Argumente im Definitionsbereich Listenterme sind. D.h. wir nehmen z.B. an, daß das Kommen der Kollegen von Hans eine Situation konstituiert, nicht ebenso viele Situationen, wie es Kollegen von Hans gibt.

/7.6./ Nach /7.2./ und /7.4./ sind Terme der Form

$$\begin{aligned} & (\text{LAMBDA } x^{s_i} (F t_1^{s_{i_1}} \dots t_{n-1}^{s_{i_{n-1}}} x^{s_i})) = \\ & (F t_1^{s_{i_1}} \dots t_{n-1}^{s_{i_{n-1}}}) \end{aligned}$$

wo F ein zuordnend singuläres Operationssymbol ist und t_1, \dots, t_{n-1} Individuenterme sind, Individuenterme.

Beispiel:

$$\begin{aligned} (7) (\text{LAMBDA } x^{\text{natper}} (\text{VATER HANS } x^{\text{natper}})) &= \\ (\text{VATER HANS}) & \\ \text{'der Vater von Hans'}. & \end{aligned}$$

/7.7./ Nach /7.2./ und /7.4./ sind Terme der Form

$$\begin{aligned} & (\text{LAMBDA } x^{s_i} (F t_1^{s_{i_1}} \dots t_{n-1}^{s_{i_{n-1}}} x^{s_i})) = \\ & (F t_1^{s_{i_1}} \dots t_{n-1}^{s_{i_{n-1}}}) \end{aligned}$$

wo F ein zuordnend singuläres Operationssymbol ist und mindestens einer der Terme t_1, \dots, t_{n-1} ein Listenterm ist, Listenterme. Wir gruppieren diese Listenterme den relationalen Listentermen zu.

Beispiel:

$$\begin{aligned} (8) (\text{LAMBDA } x^{\text{natper}} (\text{VATER } (\text{LAMBDA } y^{\text{natper}} (\text{KOLLEGE HANS} \\ y^{\text{natper}})) x^{\text{natper}})) &= \\ (\text{VATER } (\text{KOLLEGE HANS})) & \\ \text{'die Väter der Kollegen von Hans'}. & \end{aligned}$$

Semantische Erklärung:

Der vorliegende Fall von Termeinbettung - sie kann durch rekursive Anwendung der Termbildungsregeln für LAMBDA-Terme zu mehrfachen Schachtelungen führen - stellt semantisch eine Generalisierung des einfachen Falls der Termbildung dar:

Wenn wir mit

(KOLLEGE HANS)

die Elemente einer Menge N

$$N = \{x \mid (\text{KOLLEGE HANS } x)\}$$

bezeichnen, so bezeichnen wir mit

(VATER (KOLLEGE HANS))

die Elemente der Vereinigungsmenge M

$$M = \bigcup_{x \in N} (\text{VATER } x)$$

wobei für alle $x \in N$ gilt:

(VATER x) = {y | (VATER x y)}

/8/ Wenn t^{s_i} ein relationaler Listenterm (nach /7.3./ oder /7.7./) ist, so ist

(JOTA t^{s_i})

ein Individuenterm der Sorte s_i .

Beispiel:

(9) (JOTA (KOLLEGE HANS))

'der Kollege von Hans'

(10) (JOTA (VATER (KOLLEGE HANS)))

'der Vater der Kollegen von Hans'

Semantische Erklärung:

Die Präfigierung mit JOTA zeigt an, daß es sich bei dem Denotat der präfigierten Liste um eine einelementige Liste handelt. Einelementige Listen sind per definitionem identifizierbar mit dem in ihnen enthaltenen Element. Anders als bei Termen nach /7.5./ und /7.6./ ist die Singularität des Bezeichneten nicht eine notwendige Folge der Eindeutigkeit des Prädikatszeichens, sondern ein zufälliges Faktum der beschriebenen Welt.

/9/ Wenn $t_1^{s_1}, \dots, t_n^{s_n}$ Terme jeweils der Sorten s_1, \dots, s_n sind, dann ist

$$(\text{LISTE } t_1^{s_1} \dots t_n^{s_n})$$

ein Listenterm (aufgezählter Listenterm) der Sorte

$$s_h \text{ gdw für alle } i \in \{1, \dots, n\}$$

$$s_i \leq s_h$$

Beispiele:

(11) (LISTE HANS FRANZ)

'Hans und Franz'

(12) (LISTE HANS (KOLLEGE FRANZ))

'Hans und die Kollegen von Franz'

Semantische Erklärung:

Durch einen aufgezählten Listenterm bezeichnen wir die Elemente der Vereinigungsmenge derjenigen Mengen, deren Elemente wir durch die Argumentterme bezeichnen.

/10/ Wenn $t_1^{s_i}$ und $t_2^{s_j}$ relationale oder aufgezählte Listenterme sind, oder Listenterme gemäß /10/ oder /11/ jeweils der Sorten s_i und s_j , dann ist

$$(\text{ET } t_1^{s_i} t_2^{s_j})$$

ein Listenterm der Sorte s_h gdw

$$(s_i = s_h \wedge s_i \leq s_j) \vee (s_j = s_h \wedge s_j \leq s_i)$$

Beispiel:

(13) (ET (KOLLEGE HANS) (FREUND FRANZ))

'diejenigen, die sowohl Kollegen von Hans als auch Freunde von Franz sind'.

Semantische Erklärung:

Durch einen ET-Term bezeichnen wir die Elemente der Schnittmenge derjenigen Mengen, deren Elemente durch die Argumentterme bezeichnet werden.

Bemerkung zur Sortenrestriktion:

Der ET-Term übernimmt die Sortenspezifikation des Argumentterms mit der untergeordneten Sorte. Dies entspricht dem na-

türlichsprachlichen Gebrauch: Bei durchschnittsbildenden Adjektivphrasen z.B. kann das Adjektiv einer generelleren Sorte zugehören, das Nomen einer untergeordneten, spezielleren. Die gesamte Nominalphrase mit attributivem Adjektiv hat die speziellere Sorte des Nomens:

*die guten Menschen \Leftrightarrow diejenigen, die sowohl gut
als auch Menschen sind.*

/11/ Wenn $t_1^{s_i}$ und $t_2^{s_j}$ Listenterme nach /7.2./, /7.3./, /7.7./, /9/, /10/ oder /11/ jeweils der Sorten s_i und s_j sind, dann ist

(COMPL $t_1^{s_i}$ $t_2^{s_j}$)

ein Listenterm der Sorte s_j gdw $s_i \leq s_j$.

Beispiel:

(14) (COMPL (KOLLEGE HANS) (FREUND FRANZ))

'diejenigen Freunde von Franz, die nicht Kollegen von Hans sind'.

Semantische Erklärung:

Wenn wir mit $t_2^{s_j}$ die Elemente einer Menge M bezeichnen und mit $t_1^{s_i}$ die Elemente einer Menge N, dann bezeichnen wir mit (COMPL $t_1^{s_i}$ $t_2^{s_j}$) die Elemente der Menge M-N.

/12/ Wenn t^{s_i} ein Listenterm nach /7.2./, /7.3./, /7.7./, /9/, /10/, oder /11/ der Sorte s_i ist und QU ein Quantifikator von KS, dann ist

(QU t^{s_i})

ein quantifizierter Listenterm der Sorte s_i . Quantifikatoren sind die natürlichen Zahlen und die Symbole

ALL, EIN, VIEL, MEIST.

Beispiel:

(15) (3 (KOLLEGE HANS))

'drei Kollegen von Hans'

(16) (EIN (KOLLEGE HANS))

'einige Kollegen von Hans'

Semantische Erklärung:

Aussagen, die quantifizierte Listenterme enthalten, werden durch Elemente von Untermengen derjenigen Menge erfüllt, deren Elemente durch den Listenterm selbst bezeichnet werden²⁷. Die Kardinalität der Untermenge ist im Falle der natürlichen Zahlen exakt spezifiziert, im Fall von EIN, VIEL, MEIST jedoch vage und durch detailliertere semantische Untersuchungen festzulegen.

/13/ Wenn t^s_i ein Listenterm der Sorte s_i ist, so ist

(RESPEKTIV t^s_i) oder kürzer (RESP t^s_i)

ein Listenterm der Sorte s_i .

Kommentar:

Die Präfigierung eines Listenterms mit RESPEKTIV betrifft immer mehrere Vorkommen dieses Listenterms in einer Formel. Sie zeigt an, daß die Formel nur wahr wird, wenn für alle Vorkommen des Listenterms jeweils das gleiche Element derjenigen Menge eingesetzt wird, deren Elemente durch den Listenterm bezeichnet werden.

D.h., wo P ein dreistelliges, Q und R zweistellige Prädikatsymbole und t und s Terme sind (wobei die Sortenspezifikation unberücksichtigt bleiben kann), ist

$(P (RESP(Q t) (R (RESP(Q t)))s)$

äquivalent mit

$\forall x ((Q t x) \rightarrow (P x (R x)s))$

RESPEKTIV kann somit als Wiedergabe des natürlichsprachlichen Reflexiv- oder Possessivpronomens dienen, z.B.

(WASCH (RESP (LISTE HANS FRANZ) (RESP (LISTE HANS FRANZ)))
'Hans und Franz waschen sich'

(WASCH (RESP (LISTE HANS FRANZ) (HAND (RESP (LISTE HANS FRANZ))))²⁸
'Hans und Franz waschen ihre Hände' bzw.
'Hans und Franz waschen sich die Hände'.

2.2.4. Korrespondenzen zwischen einigen Typen von NS-Nominalphrasen und KS-Termen

Im folgenden werden die KS-Übersetzungen wichtiger Nominalphrasentypen präsentiert. Damit sollen die Ausdrucksmöglichkeiten, die in KS für den referentiellen Bezug auf Objekte vorhanden sind, mit den entsprechenden Möglichkeiten des Deutschen verglichen werden.

In Abschnitt 2.1. wurde auf die Bedeutung solcher referierender Ausdrücke unterschiedlichster Art für die Interaktionssituation des Informationssystems hingewiesen.

Da die Übersetzung von Nominalphrasen (NP) in KS mit der logischen Funktion der NP variiert, nicht mit ihrer NS-Ausdrucksseite, gibt es keine eindeutigen ausdrucksseitigen Zuordnungen. Wir orientieren uns an der Form der NS-NP als Ordnungsprinzip.

- a1 der Nukleus der NP ist ein Eigenname, der in KS als Individuenkonstante übersetzt wird

der Hans $\xleftrightarrow{\text{ü}}$ HANS

die Firma G+L $\xleftrightarrow{\text{ü}}$ G+L

- a2 der Nukleus der NP ist ein Nomen, das in KS als zuordnend singuläres Operationszeichen übersetzt wird, und die Attribute sind ebenfalls definite singularische NP bzw. Nomen

der Vater von Hans $\xleftrightarrow{\text{ü}}$ (LAMBDA x^{natper}
(VATER HANS x^{natper})) =
(VATER HANS)

- a3 der Nukleus der NP ist ein Nomen, das in KS als strikt singuläres Operationszeichen übersetzt wird

die Ankunft der Brüder von Hans am 24.12.77 in Mannheim $\xleftrightarrow{\text{ü}}$
(LAMBDA x^{sit} (ANKOMM (BRUDER HANS) MANNHEIM 77/12/24 x^{sit}))
= (ANKOMM (BRUDER HANS) MANNHEIM 77/12/24)

Dies ist der Fall der Übersetzung von deverbativen Nominalisierungen.

- a4 der Nukleus der NP ist ein Nomen, das in KS als Relationszeichen übersetzt wird: Jotapräfигierter relationaler Listenterm

der Bruder von Hans $\xleftrightarrow{\text{ü}}$ (JOTA (LAMBDA x^{natper}
(BRUDER HANS x^{natper}))
= (JOTA (BRUDER HANS))

- a5 der Nukleus der NP ist ein Nomen, das in KS als zuordnend singuläres Operationszeichen übersetzt wird, und mindestens ein Attribut ist eine pluralische NP: Jotapräfигierter Listenterm

- der Vater der Kollegen von Hans* $\xleftrightarrow{\text{ü}}$ (JOTA (VATER
(KOLLEGE HANS)))
- b pluralische definite NP in der Funktion der Kennzeichnung von Elementen von Mengen: Repräsentation durch KS-Listenterme
- b1 der Nukleus der NP ist ein Nomen, das in KS als Relationszeichen übersetzt wird
- die Brüder von Hans* $\xleftrightarrow{\text{ü}}$ (BRUDER HANS)
- b2 der Nukleus der NP ist ein Nomen, das in KS als zuordnend singuläres Operationszeichen übersetzt wird, und mindestens ein Attribut ist eine pluralische NP bzw. Aufzählung
- die Gatten der Schwestern von Hans* $\xleftrightarrow{\text{ü}}$ (GATTE (SCHWESTER HANS))
- die Gatten von Emma und Frieda* $\xleftrightarrow{\text{ü}}$ (GATTE (LISTE EMMA FRIEDA))
- c singularische indefinite NP: Repräsentation durch Listenterm, der durch die natürliche Zahl 1 quantifiziert ist
- ein Kollege von Hans* $\xleftrightarrow{\text{ü}}$ (1 (KOLLEGE HANS))
- d pluralische indefinite NP: Repräsentation durch Listenterm, der durch EIN quantifiziert ist
- Kollegen von Hans* $\xleftrightarrow{\text{ü}}$ (EIN (KOLLEGE HANS))
- einige Kollegen von Hans* $\xleftrightarrow{\text{ü}}$ (EIN (KOLLEGE HANS))
- e restriktive Relativsätze: Repräsentation durch LAMBDA-Terme
- e1 ohne Bezugs-NP
- diejenigen, die am 24.12.77 in Mannheim angekommen sind*
- $\xleftrightarrow{\text{ü}}$ (LAMBDA x^{natper} (EXIST x^{sit} (ANKOMM x^{natper} MANNHEIM
77/12/24 x^{sit})))
- e2 mit Bezugs-NP
- die Kollegen von Hans, die am 24.12.77 in Mannheim angekommen sind*
- $\xleftrightarrow{\text{ü}}$ (LAMBDA x^{natper} (UND (KOLLEGE HANS x^{natper})
(EXIST x^{sit}
(ANKOMM x^{natper} MANNHEIM
77/12/24 x^{sit}))))

- f Nominalphrasen mit attributivem Adjektiv, das extensional verstanden wird: LAMBDA-Term aus UND-verknüpfter Formel

die freundlichen Kollegen von Hans

$$\begin{aligned} \longleftrightarrow & (\text{LAMBDA } x^{\text{natper}} (\text{UND } (\text{FREUNDLICH } x^{\text{natper}}) \\ & \quad (\text{KOLLEGE HANS } x^{\text{natper}}))) = \\ & (\text{ET } (\text{FREUNDLICH}) (\text{KOLLEGE HANS})) \end{aligned}$$

Die Einbettung der verschiedenen Termarten als Argumente einer Formel von KS soll durch das abschließende Beispiel veranschaulicht werden:

Die Kollegen von Hans und die Gatten derjenigen Schwestern von Fritz, die am 24.12.77 in Mannheim angekommen sind, sind befreundet.

$$\begin{aligned} \longleftrightarrow & (\text{FREUND } (\text{LAMBDA } x^{\text{natper}} (\text{KOLLEGE HANS } x^{\text{natper}})) \\ & \quad (\text{LAMBDA } y^{\text{natper}} (\text{GATTE } (\text{LAMBDA } z^{\text{natper}} \\ & \quad \quad (\text{UND } (\text{SCHWESTER FRITZ } z^{\text{natper}}) \\ & \quad \quad (\text{EXIST } x^{\text{sit}} \\ & \quad \quad (\text{ANKOMM } z^{\text{natper}} \\ & \quad \quad \text{MANNHEIM 77/12/24 } x^{\text{sit}})))) \\ & \quad y^{\text{natper}}))) \end{aligned}$$

2.3. Schlußbemerkung

Die Konstruktsprache ist in ihrem gegenwärtigen Entwicklungsstand eingeschränkt auf die Formulierung von Propositionen, die in einem Frage-Antwort-Dialog des faktischen Diskurses Anwendung finden.

Für diese Propositionen und ihre Teilausdrücke ist eine modelltheoretische Semantik definierbar, die hier jeweils für den Bereich der Terme informal angedeutet wurde.

In dem konkreten Verwendungszusammenhang eines Informationssystems IS hat die Datenbasis des IS die Funktion eines Modells für

eine gewisse Menge von Ausdrücken einer KS-Einzelsprache. Dieses Modell im Sinne der Modelltheorie ist eine Menge von Objekten und eine Menge von Relationen zwischen diesen Objekten.

Die Propositionen der KS-Einzelsprache, die im IS verwendet werden, sind wahr bezüglich dieses Modells, wenn sie Relationen beschreiben, die in diesem Modell bestehen.

Der Gebrauch von KS-Propositionen in den Sprechakten Aussage oder Frage kann durch die Präfigierung der Operatoren '.' (Aussage) und '?' (Frage) repräsentiert werden:

(.(EXIST x^{sit} (KOMM HANS MANNHEIM 77/12/24 x^{sit})))
'Hans kommt am 24.12.77 nach Mannheim.' im Sinne der Aussage 'es ist wahr, daß Hans am 24.12.77 nach Mannheim kommt.'

(?(EXIST x^{sit} (KOMM HANS MANNHEIM 27/12/24 x^{sit})))
'Kommt Hans am 24.12.77 nach Mannheim?'

Die Bedeutung dieser Operatoren kann jedoch im Rahmen der rein extensionalen modelltheoretischen Interpretation nicht direkt repräsentiert werden. Vielmehr werden sie durch das IS pragmatisch interpretiert, indem sich das IS entsprechend diesen Indikatoren verhält. Wird der Indikator '.' verwendet, so erweitert das System sein Modell um eine neue Instanz einer Relation zwischen Objekten, beziehungsweise es läßt sein Modell unverändert, wenn diese Instanz bereits vorhanden ist. Wird der Indikator '?' verwendet, so prüft das IS die Gültigkeit der enthaltenen Propositionen gegenüber dem Modell der Datenbasis ab.

Dieser Interpretationsvorgang wird im einzelnen im Beitrag von BERRY-ROGGHE und DILGER in diesem Band beschrieben. Dort wird der Aspekt der Algorithmisierung der Interpretation von KS-Fragen bezüglich der Datenbasis betont.

Anmerkungen

- 1 Dies trifft u.a. für die Arbeiten von Schank und Wilks zu: Die interlinguale Repräsentationssprache in Wilks, 1973 und seinen übrigen Arbeiten ist aus "Elementen" aufgebaut, die "Sinne von englischen Wörtern" (S. 123) ausdrücken.
Die Zeichen für diese Sinne sind wiederum englische Wörter, z.B. FLOW mit dem "Sinn" 'moving as liquids do'.
Wilks argumentiert einerseits, daß eine prädikatenlogische Sprache für die Zwecke der interlingualen Repräsentation bei der maschinellen Übersetzung zu aufwendig sei. Andererseits hält er generell modelltheoretisch orientierte Repräsentationssprachen für KI-unangemessen, da der Begriff der Wahrheitsbedingung nicht "computable" sei. Vgl. dazu vor allem auch Wilks, 1976b. Zu einer gegenläufigen Argumentation vgl. Hobbs/Rosenschein, 1977.
- 2 Der Begriff des Handlungszusammenhangs soll die Gesamtheit intentionaler Interaktionen bezeichnen, in die eine spezielle Handlung/Sprachhandlung eingebettet ist. Er bleibt hier notwendigerweise vage, seine Definition würde die Festlegung von Interaktionstypen voraussetzen. Ein Beispiel für einen Handlungszusammenhang in diesem Sinne wird mit dem Handlungszusammenhang 'Informationsgewinnung' im folgenden gegeben.
- 3 L_2 hat demnach gegenüber L_1 genau die Funktion, die Montague der Sprache der intensionalen Logik gegenüber einem Fragment des Englischen zuspricht, vgl. Montague/Schnelle, S. 48, K. 5.2. Zu einer formalen Beschreibung dieses Übersetzungsbegriffs und seiner Realisierung in einem IS vgl. Wulz, in diesem Band.
- 4 Hayes, 1977 weist darauf hin, daß Repräsentationen im KI-Bereich "vision" oder "scene analysis" sprachunabhängig sein müssen und daß hier sogar häufig die Repräsentationseinheiten schwerlich in NS umsetzbar sind: "Much of what a vision program needs to represent may not be readily expressible in English (e.g. 2-dimensional patterns of light and shade)" (S. 560).
- 5 Das Affen-Banane-Problem wird in mehreren Varianten vielfach in der Literatur behandelt. Die klassische Formulierung findet sich in McCarthy, 1968.
- 6 Vgl. dazu Hendrix, 1973.
- 7 Zum automatischen Beweisen vgl. Dilger in diesem Band mit Hinweisen auf die Literatur zum Thema.
- 8 In einem modalen KI-Kalkül werden Aktionen und Veränderungen durch die Modaloperatoren "can", "cause" und "canult" darstellbar gemacht, denn die an Handlungen oder Veränderungen geknüpften Fakten sind nicht grundsätzlich oder immer wahr, sondern nur möglicherweise und in bestimmten Fällen, z.B. der Verursachung durch andere Fakten.
In McCarthy und Hayes wird anstelle des problematisch zu handhabenden Modalkalküls ein Situationskalkül vorgeschlagen:

It is possible to gain the expressive power of modal logic without using modal operators by constructing an ordinary truth-functional logic which describes the multiple-world semantics of modal logic directly. To do this we give every predicate an extra argument (the

world-variable, or in our terminology, the situation variable. (...)
The resulting theory will be expressed in the notation of the situation calculus; the proposition ϕ has become a propositional fluent $\lambda s.\phi(s)$, and the 'possible worlds' of the modal semantics are precisely the situations" (S. 495).

Zum Situationenkalkül vgl. auch McCarthy, 1968, Green, 1969, McCarthy/Hayes, 1969, Hayes, 1971 und Pople, 1972.

- 9 So u.a. die Programmiersprachen PLANNER (Hewitt, 1969, Hewitt, 1972 und Sussman/Winograd/Charniak, 1971), QA4 (Rulifson/Derksen/Waldinger), Q-LISP (Reboh/Sacerdoti, 1973), CONNIVER (McDermott/Sussman, 1972).
- 10 Vgl. zu ALPHA Codd, 1971, Zu SQUARE Boyce/Chamberlin/King/Hammer, 1975.
- 11 Vgl. dazu die Kritik Sgalls: Er hält die Beschreibung des Zusammenspiels der verschiedenen Wissens Ebenen in der KI, speziell bei Winograd, 1972 für instruktiv auch für linguistisch-kommunikative Analysen, hält aber an der systematischen Separierbarkeit und unabhängigen Beschreibbarkeit des "Sprachsystems" fest.
- 12 Der Begriff des Theorems in prozeduralen Sprachen ist wohl zu unterscheiden vom logischen Theorembegriff. Er wird im folgenden Text erläutert. In den PLANNER-ähnlichen Sprachen werden "consequent theorems" und "antecedent theorems" unterschieden. Bei dem folgenden Beispiel handelt es sich um ein consequent theorem. Winograd, 1974 beschreibt die Distinktion wie folgt:

In the consequent case we say 'Here is what I want, go try to do it' while in the antecedent case, the message is 'Here is what I've just found, what can you do with it?'" (S. 55).
- 13 Hayes, 1977, 559. Ähnlich schon Hayes, 1974, 64 ff.
- 14 Hayes, 1977 weist darauf hin, daß der Objektbereich 'Strategie' durch Aussagen über Beweise, ihr Erfolgtsein bzw. Nicht-Erfolgtsein z.T. erfaßt und in einer Logiksprache beschrieben werden kann.
- 15 Winograd, 1975, 186.
- 16 Auf Wittgenstein als Wegbereiter des prozeduralen Ansatzes beruft sich Wilks, 1976b.

Die Assoziation zu Wittgenstein liegt immerhin näher als der Vergleich der "imperativen" Semantik Winograds und der Prozeduralisten mit der formalen Semantik der Montague-Grammatik. Sgall, 1976, 158f. stellt diesen Vergleich an und beruft sich auf die Auffassung von Bedeutungen als Funktionen von Entitäten bestimmter Art in Entitäten bestimmter Art. Der mathematische Funktionsbegriff hat jedoch m.E. wenig mit dem anthropomorphen Begriff des "Imperativen" zu tun, ganz davon abgesehen, daß ja die prozedurale Beschreibung zunächst ein uninterpretierter Formalismus ist.
- 17 Schubert, 1976, 166.
- 18 Z.B. Simmons/Chester, 1977; dort weitere Literatur.

- 19 Schubert, 1976, 175 ff., besonders 176: "This first of all requires a distinction between existentially and universally quantified nodes. A simple method is the use of solid lines for existentially quantified concept nodes and broken lines for universally quantified nodes."
- 20 Vgl. Hayes, 1974, 66.
- 21 Schubert, 1976, 164.
- 22 Diese Idee lag bereits dem ersten Entwurf eines semantischen Netzes bei Quillian, 1968 zugrunde, der sein Netz als "semantic memory" bezeichnete. Weitere Arbeiten in dieser Richtung sind neben den Arbeiten Schanks die von Rumelhart/Lindsay/Norman, 1972, Rumelhart/Norman, 1973 oder Abelson, 1973.
- 23 Allerdings werden die KS-Repräsentationen von NS-Verben mit einer speziellen Sortenstruktur ausgestattet, vgl. dazu unten.
Wortartunterscheidungen zwischen Verben, Nomina (common nouns) und Adjektiven werden bei Semantiksprachen auf kategorialer Basis, so bei Lewis, 1972, Cresswell, 1973 und Montague, 1973 - in verschiedener Weise -, durch die Zuweisung unterschiedlicher Kategorien gemacht. Dies ist entsprechend der zentralen Idee dieser Semantiksprachen notwendig, weil die syntaktische Kategorienzuordnung, die gemäß der Funktion in NS-Ausdrücken geschieht, und die semantische Kategorienzuordnung eindeutig aufeinander bezogen sind.
- 24 Zum Verhältnis natürlichsprachlicher Dialogpartner vs. 'Welt des IS' sind folgende Anmerkungen zu machen:
- a) Die Voraussetzungen, die der menschliche Dialogpartner macht, gelten immer bezüglich der Welt des IS im Sinne des abgespeicherten Modells, nicht bezüglich der Außenwelt, über die geredet wird. Denn es hat ja nur Sinn, über dasjenige Informationen gewinnen zu wollen, wovon bekannt ist, daß es im Modell überhaupt seinen Platz hat.
 - b) Wir müssen annehmen, daß die Welt des IS 'geschlossen' ist, d.h. daß alle relevanten Informationen über den Anwendungsbereich, sofern es sich um eine im IS erfaßte Problemstellung handelt, abgespeichert sind. Wir können daher im Anwendungsbereich 'Abwasserkontrolle' annehmen, daß es keine Proben innerhalb des gewählten regionalen Gebietes gibt außer den im IS abgespeicherten. Negierte Aussagen müssen daher bei der "closed world"-Hypothese nicht abgespeichert werden. Was zu einem im IS präsenten Thema nicht repräsentiert ist, ist nicht der Fall.
- 25 Vgl. die Beiträge von Dilger und Wulz in diesem Band. Zum Gebrauch sortierter Kalküle im Rahmen der KI vgl. Hayes, 1969, 539, Hayes, 1971, 500 und Sandewall, 1970. Zu einem logischen Sortenkalkül, der auch linguistische Aspekte berücksichtigt, vgl. Thomason, 1972.
- 26 Diese Repräsentation setzt die Annahme voraus, daß die Zeit aus diskreten Zeitpunkten, die als Individuen auffaßbar sind, besteht. Diese Auffassung mag für praktische Zwecke in einem IS angehen. Geht man von der Vorstellung der Zeit als Kontinuum aus, so ist jede Zeitangabe als Angabe eines Intervalls aufzufassen, dem als Extension eine unendlich große Menge von in ihm enthaltenen Intervallen zukommt. Eine solche Zeitangabe ist als Listenterm zu repräsentieren.

- 27 Dabei ist selbstverständlich zu berücksichtigen, daß quantifizierte Ausdrücke ("quantifier phrases") keine Individuen bezeichnen, vgl. Montague, 1973.
- 28 Neben RESPEKTIV ist beim weiteren Ausbau von KS auch eine Entsprechung für das natürlichsprachliche *gegenseitig* vorzusehen, das die nicht-parallele Abarbeitung von Listen vorschreibt. Beispiel: *Hans und Franz waschen sich gegenseitig die Hände.*

AUTOMATISCHE SYNTAKTISCHE ANALYSE DES DEUTSCHEN MIT ÜBERGANGSNETZWERKEN

1. Aufgabenstellung
2. Formale Bestimmungsgrößen für einen Parser
 - 2.1. Transparenz und Flexibilität
 - 2.2. Verarbeitbare Grammatiktypen
 - 2.3. Top-down - bottom-up
 - 2.4. Prädiktive Suche, parallele Suche, Backtracking
 - 2.5. Ergebniserzeugung
3. Linguistische Anforderungen an einen Parser für das Deutsche
 - 3.1. Phrasenstrukturgrammatiken
 - 3.2. Variable Satzgliedfolge
 - 3.3. Wiederholbare gleichgeordnete Konstituenten
 - 3.4. Morphologische Merkmale und ihre Berechnung
 - 3.5. Diskontinuierliche Konstituenten
 - 3.6. Schlußfolgerungen
 - 3.7. Ausgeklammerte Probleme
4. Übergangsnetzwerke: Allgemeine Charakterisierung
 - 4.1. Einfache Übergangsnetzwerke
 - 4.2. Erweiterte Übergangsnetzwerke
 - 4.3. Eine Netzwerksprache für die syntaktische Analyse des Deutschen in PLIDIS
 - 4.3.1. Kantentypen
 - 4.3.2. Aktionen
 - 4.3.3. Formen
 - 4.3.4. Tests

5. Beispiele für Problemlösungsstrategien in Übergangsnetzwerken
 - 5.1. Überprüfung und Berechnung morphologischer Bedingungen
 - 5.2. Steuerung der Übergänge durch Kantenbedingungen
 - 5.3. Diskontinuierliche Konstituenten
 - 5.4. Ergebniserzeugung
 - 5.5. Vermeidung von Backtracking
6. Zusammenfassende Charakterisierung der Leistungsfähigkeit von erweiterten Übergangsnetzwerken
7. Satzanalyse mit Übergangsnetzwerken in PLIDIS

Anmerkungen

1. Aufgabenstellung

Ein System zur automatischen syntaktischen Analyse, kurz Parser genannt, hat folgende Aufgaben zu erfüllen:

- Für beliebige Eingabeäußerungen ist festzustellen, ob sie mit den Regeln einer bestimmten, gegebenen Grammatik beschreibbar sind oder nicht. Es sollen alle und nur die Äußerungen akzeptiert werden, auf die die Regeln der Grammatik zutreffen. Wenn eine Äußerung mit der Grammatik auf mehrere Weisen beschreibbar ist, dann muß der Parser in der Lage sein, alle Beschreibungsmöglichkeiten zu finden. (Vorauszusetzen ist dabei natürlich, daß die Menge der Regeln und die Menge der Analysemöglichkeiten immer endlich sind.)
- Wenn eine Eingabeäußerung mit einer Grammatik erfolgreich analysiert worden ist, muß als Ergebnis eine Strukturbeschreibung dieser Äußerung zurückgegeben werden. Andernfalls muß die Meldung erfolgen, daß die Äußerung mit der gegebenen Grammatik nicht beschreibbar ist.

Ganz allgemein gesehen, muß im Falle von natürlichen Sprachen eine vollständige Satzanalyse darin bestehen, daß zuerst jedes Wort auf seine morphologischen Eigenschaften untersucht wird und daß daran anschließend untersucht wird, in welcher Weise die morphologisch identifizierten Wörter sich zu größeren Einheiten zusammenfassen lassen. Die beiden Aufgaben "morphologische Identifizierung" und "syntaktische Gruppierung" lassen sich leicht voneinander trennen; im Ablauf einer Analyse handelt es sich dann um zwei getrennte aufeinanderfolgende Schritte. Auch im folgenden wird eine solche Modularisierung vorausgesetzt (s.a. Morphosyntaktische Analyse in einem Frage-Antwort-System, in diesem Band). Wenn von "syntaktischer Analyse" gesprochen wird, dann ist damit stets "syntaktische Analyse einer Kette morphologisch identifizierter Wörter" gemeint. Formal gesehen bleibt sich die Problematik mit dieser Spezifizierung jedoch gleich. Denn es spielt auf dieser Ebene keine Rolle, ob in einem Durchgang zuerst für alle Wörter eines Satzes die möglichen morphologischen Beschreibungen ermittelt werden, oder ob dies für jedes einzelne Wort erst in

dem Moment geschieht, wo das Wort für seine syntaktische Einordnung eingelesen wird.

2. Formale Bestimmungsgrößen für einen Parser

Die in 1. erwähnte Aufgabenstellung läßt sich auch als die Aufgabe umschreiben, einen Algorithmus (d.h. eine Folge von ausführbaren Befehlen) zu konstruieren, der die in einer Grammatik gegebenen Regeln auf eine Eingabeäußerung "anwendet" und dabei die gewünschten Resultate erzielt. Während eine übliche Grammatik als statisches System von Gesetzen interpretiert werden kann, muß also ein Parser demgegenüber prozeduralen Charakter haben, Befehle ausführen.

Die allgemeine Aufgabenstellung läßt sich nun mit einer großen Anzahl unterschiedlicher Konzeptionen und Methoden lösen. Je nach Vorgehensweise ergeben sich verschiedene Systeme mit unterschiedlichen Eigenschaften. Für welches System man sich entscheidet, hängt zum großen Teil von der Bewertung der einzelnen Aspekte ab. Gar nicht gehen wir im folgenden auf das Problem der Speicher- und Rechenzeitanforderungen ein.

2.1. Transparenz und Flexibilität

Als erste Kriterien für die Beurteilung eines Analysesystems seien Transparenz und Flexibilität genannt. Beides sind zwar nicht unbedingt Kriterien, die die Leistungsfähigkeit eines Parsers an sich beschreiben, sie beziehen sich eher auf die Handhabbarkeit durch den Konstrukteur eines Analysesystems. Diese spielt jedoch in der Praxis eine nicht zu unterschätzende Rolle. Bei der Entwicklung und später bei der Verwendung im praktischen Gebrauch ist es wichtig, daß die vom Konstrukteur getroffenen Entscheidungen und die damit verknüpften Konsequenzen ohne Aufwand leicht zu erkennen sind.

Transparenz und Flexibilität sind allerdings Kriterien, die sich

leichter intuitiv beschreiben als formal definieren lassen. Wenn wir einzelne spezifische Merkmale angeben wollen, dann können wir sagen, daß ein Analysesystem transparent genannt werden kann, wenn eine Operation, die in bezug auf die Aufgabenstellung und die Strategie des verwendeten Verfahrens einfach und elementar ist, ebenso einfach ausgedrückt werden kann, wenn die Wirkung der Operation in ihrem funktionalen Kontext anhand der Darstellung selbst ersichtlich wird; wenn das Verfahren eine Organisation in sinnvolle und überschaubare Teilfunktionen zuläßt; und schließlich, wenn in bezug auf die Strategie des Verfahrens variable Größen auch als solche in Erscheinung treten.

Die Kriterien Transparenz und Flexibilität bekommen unter anderem bei der Frage nach dem Zusammenhang von Analysealgorithmus und grammatischem Regelwerk Relevanz. Dieses Problem kann in verschiedener Weise gelöst werden.

Eines der möglichen Verfahren besteht darin, die gegebenen Grammatikregeln direkt in einem Analysealgorithmus darzustellen. Eine Phrasenstrukturregel wie (1) wäre so beispielsweise als Operation "NP" wie in (2) auszudrücken:

(1) NP \rightarrow DET N

(2) NP: 1. Lies das nächste Wort.

Wenn es die Kategorie DET hat, gehe nach 2.

Andernfalls gib als Wert "Mißerfolg" zurück.

2. Lies das nächste Wort.

Wenn es die Kategorie N hat, gib als Wert "Erfolg" zurück.

Andernfalls gib als Wert "Mißerfolg" zurück.

Eine zweite Möglichkeit besteht darin, zusätzlich zu gegebenen Grammatikregeln einen Algorithmus zu formulieren, der in genereller Weise Grammatikregeln auf Eingabeäußerungen anwendet. Ein solcher Algorithmus enthält Anweisungen, aufgrund derer in systematischer Weise Regeln beliebiger Art auf Eingabeäußerungen angewendet werden, wobei z.B. geprüft wird, ob ein Wort einer gewünschten Kategorie angehört, oder ob aufgrund vorangegangener Analysen und einer neuen Regel zwei Konstituenten zu einer sol-

chen höheren Typs vereinigt werden können. Die speziellen Kategorien und Regeln erscheinen dabei nicht primär im Algorithmus selbst, sondern werden im Lauf der Analyse aus der Grammatik eingelesen.

Das erste Verfahren ist zwar gegenüber dem zweiten sparsamer und schneller, da dabei weniger zusätzliche Berechnungen nötig sind, und sämtliche grammatischen Daten dort, wo sie benötigt werden, explizit bereits vorhanden sind. Im zweiten Verfahren müssen zusätzliche Kontrollen über die angewendeten Regeln und Regelteile und Einleseoperationen vorgenommen werden. Jedoch entspricht das zweite Verfahren den Forderungen nach Transparenz und Flexibilität eher als das erste. Da der Analysealgorithmus im Prinzip eine allgemeine Analysestrategie darstellt und für seine Konstruktion der Inhalt einzelner Regeln keine Rolle spielt, sind die Regeln selbst leicht je nach den besonderen Anforderungen veränderbar. Im Falle des ersten Verfahrens würden Regeländerungen zu möglicherweise folgeschweren Veränderungen in der Analyseprozedur führen. Wenn die Regeln von der Analyseprozedur losgelöst formuliert werden können, bedeutet dies auch für den Konstrukteur einer Grammatik eine bedeutende Erleichterung.

Die Forderung nach Transparenz führt also im Normalfall zur Entscheidung zugunsten eines Parsers, in dem Analysealgorithmus bzw. Formulierung einer Analysestrategie und Grammatikregeln systematisch voneinander getrennt sind.

Ein weiteres Anwendungsbeispiel für die Kriterien Transparenz und Flexibilität bietet das Problem der Verknüpfbarkeit verschiedener Operationen in einem Analysesystem. Wir müssen ja davon ausgehen, daß in einem solchen Analysesystem neben Operationen zur Interpretation grammatischer Regeln noch andere Operationen in Anwendung kommen, wie z.B. Test- und Ausgabeoperationen. Im allgemeinen ist mit der Koordination der verschiedenen Operationsanwendungen ein gewisser Kontroll- und Steueraufwand verbunden. Einerseits soll dieser Kontrollapparat möglichst einfach gehalten werden, andererseits sollen viele Operationen von Fall zu Fall speziell eingesetzt werden können. Um zwischen diesen widersprüchlichen Forderungen ein Gleichgewicht zu finden, muß eine Unterschei-

dung zwischen generellen Strategien und besonderen, an beliebiger Stelle einsetzbaren Funktionen getroffen werden. Die ersteren sollen so formuliert sein, daß, wenn einmal die generelle Strategie festgelegt ist, der Konstrukteur eines Parsers sich weiter nicht mehr darum zu kümmern braucht; die besonderen Operationen sollen demgegenüber so flexibel wie möglich an beliebiger Stelle in einer Analyse einsetzbar sein; dabei ist gleichwohl wünschenswert, daß außer der Steuerung der unmittelbaren Anwendung keine weiteren Kontrollfunktionen mehr angegeben werden müssen.

2.2. Verarbeitbare Grammatiktypen

Wie bereits angesprochen, müssen konkrete Analysealgorithmen auf allgemeine formale Eigenschaften der zugelassenen Grammatikregeln Bezug nehmen. In vielen Fällen werden bestimmte Einschränkungen hinsichtlich des zugelassenen Regeltyps gemacht, um den interpretierenden Algorithmus zu vereinfachen und effizienter machen zu können. Z.B. werden für den Predictive Analyzer von S. KUNO und A.G. OETTINGER (vgl. KUNO, 1966) Regeln von der in (3) definierten Form (sogenannte "Standard-Form-Regeln") vorausgesetzt:

- (3) $Z \rightarrow c Y_1 \dots Y_m : m \geq 0$
 $Z, Y_1 \dots Y_m$ sind beliebige grammatische Kategorien
 c ist eine präterminale Kategorie
 (eine Wortklasse)

Der IC-Parser von J.J. ROBINSON (ROBINSON, 1965) seinerseits ist beschränkt auf binäre Phrasenstrukturregeln:

- (4) $A \rightarrow B C$

Für manche Parser, (s. AHO/JOHNSON/ULLMANN, 1975) wird ferner vorausgesetzt, daß die Regeln eindeutig sind, daß es also zu einem bestimmten Symbol auf der linken Seite nur eine einzige mögliche rechte Seite gibt. Andernfalls würden mehrere Analysemöglichkeiten entstehen, die mit diesem speziellen (deterministischen) Parser nicht erfaßt werden könnten.

Die Festlegung eines bestimmten Regeltyps für eine Grammatik geht natürlich Hand in Hand mit der generativen Kraft einer solchen Grammatik. In den erwähnten Fällen handelt es sich zwar stets um Sonderfälle von kontextfreien Phrasenstrukturgrammatiken; im allgemeinen gilt aber, daß ein Parser umso komplexer sein wird, je ausdruckskräftiger ein Grammatiktyp ist. So wird für kontextsensitive Grammatiken im allgemeinen kein spezieller Parser entwickelt, da dies im Vergleich zu sonst möglichen Lösungen, die auf kontextfreien Strategien basieren, allzu kompliziert wäre (s. HAYS, 1966, S. 80, HAYS, 1967, S. 148f.). Bekannt ist schließlich, daß für Transformationsgrammatiken, wie sie von Chomsky definiert worden sind, nur sehr ineffiziente Parser konstruiert werden können (vgl. KUNO, 1976).

Die Wahl eines Grammatiktyps hängt allerdings bei Parsern für die natürliche Sprache normalerweise nicht von Entscheidungen für oder gegen ein bestimmtes Parsingverfahren ab, sondern in der Hauptsache von linguistischen Erwägungen, also von Entscheidungen darüber, wie die zu behandelnde Sprache beschrieben und analysiert werden soll. Welche Anforderungen an eine Grammatik aus linguistischer Sicht zu stellen sind, soll jedoch in einem gesonderten Teil (Abschnitt 3) diskutiert werden. An dieser Stelle sei lediglich festgehalten, daß linguistische Entscheidungen Rückwirkungen auf die Gesamtkonzeption eines Parzers haben werden.

2.3. Top-down - bottom-up

Eine Satzanalyse kann nach zwei verschiedenen Strategien vor sich gehen: vom Satzganzen zu den einzelnen Wörtern ("top down") oder von den einzelnen Wörtern zum Satzganzen.

In einem top-down-Verfahren wird zunächst für ein komplexes Syntagma, z.B. eine NP, eine Regel gesucht, mit der sich eine NP in kleinere Teile zerlegen läßt, die also links von \rightarrow das Kategoriensymbol NP aufweist; es wird nachgeprüft, ob es also z.B. eine Regel von der Form $NP \rightarrow DET\ N$ gibt.

In einem bottom-up-Verfahren wird demgegenüber versucht, für bekannte bereits analysierte Elemente eine Regel zu suchen, die diese Elemente in eine übergeordnete Kategorie zusammenzufassen erlaubt. Es wird also z.B., wenn wir eine Folge von DET N gefunden haben, eine Regel gesucht, die etwa $NP \rightarrow DET\ N$ lauten könnte, also auf der rechten Seite von \rightarrow die bereits bekannten Elemente enthält.

In einem bottom-up-Verfahren wird also immer versucht, ausgehend von einer Teilkette des Eingabesatzes (die auch schon teilweise analysiert sein kann) eine Regel zu finden, deren Zielsymbole auf die Teilkette passen, und die es erlaubt, diese Teilkette zu einer höheren Konstituenten (repräsentiert durch das Ausgangssymbol der Regel) zusammenzufassen.¹

Es scheint, daß im allgemeinen die beiden Strategien gleich erfolgreich sind. Ein Problem für das top-down-Verfahren besteht jedoch darin, daß es bei einer linksverzweigenden Regel wie (5) in eine endlose Schleife gerät.

(5) R1: $A \rightarrow A\ B$

R2: $A \rightarrow C\ D$

Wenn das Symbol A im Laufe einer Analyse angetroffen wird und analysiert werden soll, wird R1 anzuwenden versucht. Hierbei wird jedoch wieder das Symbol A angetroffen, worauf wiederum R1 angewendet werden muß, usw. Eine Lösung dieses Problems kann in einer Änderung der Grammatik, durch Einführung eines Hilfssymbols oder im günstigen Fall, durch Vertauschung von R1 und R2 bestehen; letzteres kann aber im ungünstigen Fall dazu führen, daß R1 dann überhaupt nie mehr anzuwenden versucht wird.

In einem Punkt kann ein bottom-up-Verfahren wesentlich effizienter ausgebaut werden als ein top-down-Verfahren. Bei unvollständigen (z.B. elliptischen) oder sonstwie abweichenden Sätzen wird ein Parser mit top-down-Strategie die Analyse unweigerlich abbrechen müssen, wenn er auf eine "unverständliche" Teilkette stößt. Es wird auch sehr schwer sein, vom Parser in diesem Fall zu erfahren, warum die Analyse mißlungen ist. Ein bottom-up-Parser wird auch in einer solchen Situation weiterarbeiten können, wenn

er in einem bestimmten Umfang die Fähigkeit zur simultanen partiellen Analyse (s. Abschnitt 2.4.) besitzt, und also eine Analyse auf Wortebene fortsetzen kann, selbst wenn mitten im Satz ein "unverständlicher" Teilstring angetroffen worden ist. Es wird dabei auch möglich sein, mögliche analysierbare Teile anzugeben und aufzuzeigen, an welchen Stellen eine Analyse nicht weitergeführt werden kann.

2.4. Prädiktive Suche, parallele Suche, Backtracking

Wenn nicht genaue Beschränkungen über die Gesamtorganisation von Regelmengen eingeführt werden, kann es vorkommen, daß die Regelanwendungen zu mehrdeutigen Situationen führen, in denen es möglich ist, mehrere Regeln anzuwenden, ohne daß aber im Moment erkennbar wäre, welche Regel, wenn überhaupt, zu einer erfolgreichen Analyse führt. Solche Situationen entstehen (je nach Analysestrategie) z.B., wenn zwei Regeln das gleiche Ausgangssymbol (auf der linken Seite), aber verschiedene Zielsymbole (rechte Seite von \rightarrow) besitzen, oder wenn zwei Regeln teilweise identische Zielsymbolketten aufweisen.

Wenn wir z.B. in einem top-down-Verfahren die Analyse einer NP durchführen wollen und dabei die Regeln in (6) vorfinden

- (6) NP \rightarrow DET N
NP \rightarrow ZAHLWORT N
NP \rightarrow ADJ N

gibt es zunächst keinen Anhaltspunkt, welche Regel eine erfolgreiche Analyse ergeben wird. Die Fragen, die dabei entstehen, sind: Welche Regel von allen soll ich zuerst ausprobieren? Was mache ich, wenn das vorliegende Syntagma sich nicht mit einer einmal gewählten Regel analysieren läßt?

Zur Lösung dieses Problems sind zwei Arten von Vorgehensweisen entwickelt worden (wenn wir von der einfachsten Lösung, dem Verbot solcher mehrdeutigen Regelkombinationen überhaupt, absehen).

Das erste Verfahren, welches wir "prädiktive Suche" nennen wollen,² besteht darin, für eine solche mehrdeutige Situation ein

Entscheidungsverfahren vorzusehen, aufgrund dessen eine von den mehreren möglichen Regeln ausgewählt und angewendet wird. Es wird also sozusagen eine "Voraussage" gemacht, welches die zu erwartende erfolgreiche Analyse sein wird. Die Entscheidung, welche Regel als erste anzuwenden ist, kann aufgrund ganz verschiedener Kriterien gefällt werden; normalerweise wird das einfachste Vorgehen gewählt, indem man die Regeln untereinander ordnet, und dann zunächst die erste Regel in dieser Ordnung angewendet wird. Wenn es sich nun herausstellt, daß die zunächst ausgewählte Regel zu keinem Analyseergebnis führt, muß der Parser die Anwendung dieser Regel abbrechen, zum ursprünglichen Zustand zurückkehren und eine neue Regel anzuwenden versuchen. Um zu solchem sogenannten Backtracking in der Lage zu sein, muß ein Parser die Informationen zu jedem einmal erreichten Zustand wieder auffinden können, für jeden Zustand sich die Regeln merken, die er schon einmal erfolglos angewendet hat, und für jeden Zustand wissen, wie er in diesen gelangt ist.

Ein zweites Verfahren besteht in der "parallelen Suche" (s. z.B. HAYS, 1966, S. 75).³ Dabei wird so vorgegangen, daß stets alle Regeln, die in einer bestimmten Situation anwendbar sind, gleichzeitig versucht werden. Im Verlauf einer Analyse bleiben von allen Teilergebnissen die insgesamt möglichen als Gesamtergebnis übrig. Falls mehrere Analysen auf einen Satz zutreffen, erhält man alle Analysen als Ergebnis.⁴

Bei einer bottom-up-Strategie geht eine parallele Suche beispielsweise folgendermaßen vor sich: Zunächst wird für alle Wortgruppen untersucht und vermerkt, ob sie aufgrund einer Regel zu einem Konstituenten höheren Typs zusammengefaßt werden können. Eine Kette

(7) *Wollen alle Männer küssen*

könnte als Ergebnis je nach Regeln in alternative Teilketten wie (*wollen*) (*alle*) (*Männer küssen*), bzw. (*wollen*) (*alle Männer*) (*küssen*) oder (*wollen*) (*alle*) (*Männer*) (*küssen*) aufgeteilt werden. Im nächsten Schritt wird dann versucht, diese Teilketten in Ketten höherer Ordnung zusammenzufassen, also etwa in (*wollen*) ((*alle*) (*Männer küssen*)) usw.

Beide Verfahren, die prädiktive Suche mit Backtracking wie die parallele Suche, haben ihre besonderen Vorzüge und Nachteile. Mit paralleler Suche werden von vornherein alle möglichen Ergebnisse erzeugt. Wenn also ein Satz mehrdeutig ist und mehrfach analysiert werden kann, wird das Ergebnis aus mehreren Strukturbeschreibungen bestehen. Dafür wird auf dem Weg zum Ergebnis eine sehr große Zahl von Teilergebnissen aufgebaut, die später ohnehin wieder verworfen werden müssen. Besonders wenn die Regeln keine vorgegebene Länge haben, also z.B. nicht binär sind, ergibt sich eine ungeheure Zahl von unverwendbaren Zwischenresultaten, die Zeit- und Speicherbedarf unverhältnismäßig in Anspruch nehmen.

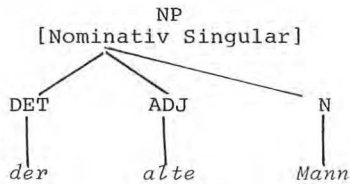
Ein prädiktives Verfahren ist in Zeit- und Raumbedarf sparsamer, da eine Struktur nur aufgebaut wird, wenn die Voraussetzungen dafür gegeben sind. Dies wird höchstens durch den Aufwand an Backtracking wieder teilweise aufgehoben.

Ferner wird bei prädiktiver Suche mit Sicherheit nur ein Ergebnis geliefert, selbst wenn mehrere Analysen möglich sein sollten. Dies kann gegenüber der Ausgabe mehrerer Ergebnisse unter bestimmten Bedingungen durchaus ein Vorteil sein. Wenn das Ergebnis weiterverarbeitet werden soll, z.B. in einem Frage-Antwort-System, stellt das Vorhandensein mehrerer Ergebnisse eher ein Problem als einen Vorteil dar.

2.5. Ergebniserzeugung

Wie in Abschnitt 1. erwähnt worden ist, ist es generell der Zweck eines Parsers, nicht nur eine Analyse von Sätzen durchzuführen und so anzugeben, ob ein Satz zur Sprache einer bestimmten Grammatik gehört oder nicht, sondern darüberhinaus dem Satz aufgrund der Analyse eine bestimmte Struktur zuzuweisen. Praktisch heißt das, daß als Ergebnis einer Analyse nicht nur eine Meldung "akzeptiert" oder "nicht akzeptiert", sondern eine um Strukturangaben erweiterte Satzdarstellung zurückgegeben werden muß. Wenn z.B. eine Kette *der alte Mann* als Nominalgruppe im Nominativ Singular, bestehend aus Determinans, Adjektiv und Nomen erkannt wurde, muß dies in

irgendeiner Weise dargestellt werden, sei es als Baumdiagramm:



sei es als Klammerstruktur (die hier verwendeten Klammerkonventionen sind willkürlich):

(NP (Nominativ Singular) ((DET *der*) (ADJ *alte*) (N *Mann*)))

Der Aufbau einer strukturierten neuen Satzdarstellung stellt eine Operation dar, die mit den bisher erwähnten Analyseoperationen noch gar nicht erfaßt ist, die also zusätzlich zu den Analyseoperationen formuliert werden muß.

Eine Vorgehensweise besteht darin, in den Interpretationsalgorithmus für Regeln selbst direkt eine allgemeine Prozedur einzubauen, die bei jeder erfolgreichen Regelanwendung ausgeführt wird, selbst aber unabhängig von speziellen Regeln ist. Wir hätten so etwa die Vorschrift, daß bei jeder Regelanwendung der angewendeten Regel entsprechend eine Baumstruktur aufgebaut wird, deren Mutterknoten das Ausgangssymbol der Regel ist, zu dem die Zielsymbole als Tochterknoten angefügt werden, und zwar in der gleichen Reihenfolge, wie sie in der Regel erscheinen.

Dadurch, daß in all diesen Fällen die Operationen zur Ergebniserzeugung vom Parser selbständig ausgeführt werden, hat der Konstrukteur einer konkreten Grammatik sich in keiner Weise um die Ergebniserzeugung zu kümmern.

Diese Allgemeinheit der Ausgabeerzeugung kann allerdings dann zum Nachteil werden, wenn bei einer Analyse die Erzeugung eines Ergebnisses in Abhängigkeit von der einzelnen angewendeten Regel oder von sonstigen Bedingungen gesteuert werden soll. So könnte man sich wünschen, daß die Reihenfolge der Konstituenten im Sinne einer Normalisierung verändert wird, daß z.B. der gesamte Verbkomplex im Ergebnis stets als erste Konstituente erscheint oder daß in der Ausgabe Formwörter usw. weggelassen oder zusammengezogen

werden usw. In einem gewissen Sinn bedeuten solche Veränderungen gegenüber den Eingabeäußerungen bereits Transformationen über einem analysierten Satz.

Die Konsequenz solcher individueller Darstellungen für einzelne Konstituenten ist, daß die Ergebniserzeugung ebenso individuell als spezielle Anweisung zu einer Regel angegeben werden muß. Für Adjektive und Nomina gilt z.B., daß sie hinter dem bisherigen Ergebnis angehängt werden, für Verben aber, daß sie allen Teilergebnissen vorangestellt werden.

Dies führt zu Formulierungsproblemen: Wie sind solche Operationen möglichst transparent darzustellen? Grammatikregeln und Ergebniserzeugungsregeln gehören zusammen und sollten auch darstellungsmäßig in möglichst engem Zusammenhang formuliert werden.

3. Linguistische Anforderungen an einen Parser für das Deutsche

Bereits in Abschnitt 2.2. ist erwähnt worden, daß die Konzeption eines Parsers entscheidend abhängig ist vom Typ der Grammatik, die bei der Analyse einer Sprache angewendet werden soll. Die Wahl des Grammatiktyps wiederum hängt, entsprechend der Zielsetzung eines Parsers, von der zu beschreibenden Sprache und der intendierten Beschreibung dieser Sprache ab. In manchen Fällen kann man sich zwar auch aushilfsweise mit einem zu primitiven Grammatiktyp behelfen, wobei man unzulängliche Beschreibungskategorien bewußt in Kauf nimmt. Eine solche bewußte Einschränkung soll im gegenwärtigen Rahmen jedoch nur in Notfällen hingenommen werden. Zweck eines Parsers soll hier ja sein, eine linguistisch möglichst adäquate Beschreibung deutscher Sätze zu geben. In der folgenden Diskussion wird deshalb zunächst weniger Wert auf Realisierbarkeit als auf Wünschbarkeit vom linguistischen Standpunkt aus gelegt. Die Vorteile einer starken Beschreibungskapazität sind erst am Schluß gegenüber den damit verbundenen Problemen der Verarbeitung abzuwägen.

3.1. Phrasenstrukturgrammatiken

Das Beschreibungsmodell par excellence für Sprachen, besonders für formale Sprachen, ist die Phrasenstrukturgrammatik. Die damit beschreibbaren Sprachen sind, wie man vom Beispiel logischer Kalküle her weiß, sowohl sehr ausdrucksstark, besonders hinsichtlich der unbegrenzten Rekursionsmöglichkeiten, wie einfach zu definieren und zu analysieren.

Bekanntlich entsprechen natürliche Sprachen, oder genauer die vom linguistischen Standpunkt aus erwünschten Beschreibungen natürlicher Sprachen, in vielen Teilen nicht den Einschränkungen einer Phrasenstrukturgrammatik. Im folgenden seien die für das Deutsche charakteristischen zusätzlich erwünschten Möglichkeiten kurz diskutiert. An dieser Stelle sei lediglich ergänzt, daß wir, selbst wenn wir eine Phrasenstrukturgrammatik einer Darstellung des Deutschen zugrundelegen, mit Vorteil eine solche ohne zusätzliche Beschränkungen benötigen, also eine Grammatik, die nicht auf binäre Regeln oder Standardform-Regeln (siehe Abschnitt 2.2.) beschränkt ist, sondern so viele Zielsymbole wie erwünscht, in beliebiger Art erlaubt.

3.2. Variable Satzgliedfolge

Wünschenswert wäre es, wenn die in wichtigen Teilen variable Satzgliedfolge im Deutschen so erfaßt werden könnte, daß nicht für jede mögliche Permutation jeweils eine besondere Regel nötig ist, sondern daß alle Permutationsfälle in einer einzigen Regel erfaßbar wären. Es ist im Deutschen praktisch automatisch vorhersehbar, daß, wenn wir einen Satz wie (8a) haben, auch eine Stellung (8b) möglich ist (bei entsprechenden Kontextbedingungen):

(8a) *Hans hat den Pudding mit dem Löffel umgerührt.*

(8b) *Hans hat mit dem Löffel den Pudding umgerührt.*

Mit einer Phrasenstrukturgrammatik brauchen wir für jede Permutation eine besondere Regel. Das heißt z.B., daß wir im allgemeinen für jede mögliche Anzahl n von nominalen und adverbialen Elemen-

ten im Mittelfeld $n!$ Regeln formulieren müssen. Dies bedeutet nicht nur einen unverhältnismäßigen Aufwand von Regeln, dadurch wird eine Grammatik insgesamt unübersichtlich.

3.3. Wiederholbare gleichgeordnete Konstituenten

Zuweilen ist es erforderlich, eine variable Menge von identischen, gleichgeordneten Zielsymbolen in einer Regel zuzulassen, also über eine Operation, die dem Kleeneschen Sternoperator entspricht, zu verfügen. Beispielsweise ist es im Deutschen oft möglich, einer Nominalgruppe eine variable Anzahl von gleichgeordneten Präpositionalnominalgruppen folgen zu lassen:

- (9a) *Der Mann mit dem Hut vor dem Haus auf der Bank vom letzten Sonntag während des Narrenumzugs ...*

Auch Adjektive können vor Nomen in variabler Anzahl als Konstituenten auf gleicher Ebene erscheinen:

- (9b) *Der große, dicke, schwarzhaarige, wulstnasige ... Boxer.*

Das bekannteste Beispiel für solche Strukturen bilden koordinierte Gruppen:

- (9c) *Hans und Fritz und Maria und Frieda ... gingen baden.*

Auch solche Regeln gehen über die starke generative Kapazität von Phrasenstrukturregeln hinaus. Um sie darzustellen, müßte man für jede Anzahl von identischen gleichgeordneten Konstituenten eine besondere Regel schreiben (so ist auch die Verwendung des Kleeneschen Sternoperators bei Phrasenstrukturregeln zu interpretieren), oder sich mit einer rekursiven binären Regel behelfen, wobei dann aber die Strukturadäquatheit verloren geht.

3.4. Morphologische Merkmale und ihre Berechnung

Das Deutsche besitzt insbesondere für den Aufbau der Nominalgruppe ein kompliziertes System morphologischer Entsprechungsregeln; die meisten Elemente in einer Nominalgruppe müssen in Kasus (4

mögliche Werte), Genus (3 mögliche Werte) und Numerus (2 mögliche Werte), die in Kombination 24 möglichen Wertekonfigurationen entsprechen, übereinstimmen. Kennt man für Grammatikregeln nur die Möglichkeit einfacher Symbole, ergibt sich daraus die Notwendigkeit, für eine Regelmenge zur Beschreibung einer Nominalgruppe 24 spezialisierte Regelmengen zu formulieren. Die Merkmalsberechnung kann noch komplizierter werden, wenn noch speziellere Bedingungen abgefragt werden müssen. Die Flexion des Adjektivs hängt z.B. vom Determinans ab; der Numerus des Verbs muß mit jenem des Subjekts übereinstimmen. Nur am Rande seien schließlich die morphologischen Bedingungen, die von Verben über ihre Ergänzungen ausgehen, erwähnt.

Wenigstens für die einfacheren Fälle solcher Problemstellungen sind in der Praxis vereinfachende Formalismen entwickelt worden. Sie bauen auf dem Prinzip auf, die Darstellung morphologischer Bedingungen von den syntaktischen Regeln zu trennen und neben den Regeln als zusätzlich zu berechnende Elemente anzugeben. Ein auf das Problem der Morphologie deutscher Nominalgruppen zugeschnittener Formalismus findet sich in KRATZER/PAUSE/v.STECHOW, 1974. In allgemeinerer Form ist dieses Vorgehen von SAGER/GRISHMAN, 1975 in der Trennung von Grammatik und "restriction language" ausgearbeitet worden. Eine solche Konzeption erlaubt es, Grammatikregeln sehr einfach, etwa in der Form von Ersetzungsregeln, anzugeben und die komplizierenden Faktoren als zusätzlich auferlegte Bedingungen über die Anwendung bestimmter Regeln anzugeben. Dieser Apparat führt allerdings über die theoretische Kapazität einer Phrasenstrukturgrammatik hinaus; dies kann auch nicht ohne Folgen auf den dazugehörigen Parser bleiben.

3.5. Diskontinuierliche Konstituenten

Es gibt im Deutschen einige Fälle von Konstituenten, deren Teile nicht unmittelbar aufeinanderfolgen, sondern durch eine variable Zahl von anderen Konstituenten voneinander getrennt sind. Der wichtigste Fall sind zusammengesetzte Verbalgruppen im Hauptsatz: Der finite Teil solcher Verben steht an zweiter Stelle im Satz,

alle ändern am Ende des Mittelfelds:

- (10a) *Hans hat gestern gegen Mittag zwei Bratwürste
gekauft.*
- (10b) *Ich möchte gerne mit Ihnen in nächster Zukunft
über eine eventuelle Gehaltserhöhung sprechen.*

Andere Fälle betreffen Nominalattribute in Nebensatzform; solche Attribute - Relativsätze, gewöhnliche Gliedsätze oder Infinitivsätze - werden, sofern die Verständlichkeit dadurch nicht beeinträchtigt wird, gerne hinter das Verb am Satzende gestellt:

- (11a) *Ich habe vorgestern einen Mann auf der Straße
gesehen, der verblüffend Julius
Cäsar geglichen hat.*
- (11b) *Ich habe meine Meinung gegenüber dir nie verheim-
licht, daß du die Linguistik
besser aufgeben würdest.*
- (11c) *Morgen wird dir dein Wunsch vielleicht
erfüllt, zwei ganze Zwetschgen-
kuchen essen zu dürfen.*

Diskontinuierliche Konstituenten sind in einer Phrasenstruktur als solche nicht darstellbar, da dabei die für Phrasenstrukturregeln notwendige Bedingung der Projektivität nicht erfüllt ist. Behelfslösungen sind höchstens in der Form möglich, daß diese Konstituenten in zwei Teilkonstituenten aufgelöst und so die Diskontinuität per Dekret aus der Welt geschafft wird.

3.6. Schlußfolgerungen

In vielen Aspekten wäre es, wie die Ausführungen in diesem Abschnitt gezeigt haben, erwünscht, zur befriedigenden Beschreibung und Analyse deutscher Sätze einen Parser und ein Grammatikmodell zu haben, die über die Kapazitäten eines Parsers mit Phrasenstrukturgrammatiken hinausgehen. Die verschiedenen Anforderungen sind verschieden leicht zu lösen und haben auch eine verschiedene Wichtigkeit. In dieser Hinsicht ist eine geeignete Verarbeitbarkeit von diskontinuierlichen Konstituenten und von morphologi-

schen Eigenschaften absolut notwendig, da eine vernünftige Interpretation deutscher Sätze niemals möglich sein wird, wenn diese Probleme nicht adäquat gelöst werden können. Weniger ins Gewicht fallen wird demgegenüber die gewünschte Art der Verarbeitung einer variablen Anzahl wiederholter gleichgeordneter Konstituenten (3.3.), da die Alternativen nicht allzu problematisch sind.

Ein besonderes Problem stellt die Forderung nach der Verarbeitung einer variablen Satzgliedfolge dar. Das grundlegende Problem besteht hier schon im Erkennen einer Konstituente als ein bestimmtes Satzglied. Dafür müssen im Prinzip nicht nur morphologische Kriterien, sondern auch semantische Kriterien überprüft werden; denn in sehr vielen Fällen sind die erschließbaren morphologischen Angaben nicht eindeutig:

Welches ist z.B. die Akkusativergänzung im folgenden Satz:

(12) *Otto schenkt Mädchen immer Rosen.*

In manchen Fällen reichen selbst semantische Angaben nicht zur Vereindeutigung; auch Kontextbedingungen müßten zur Auflösung beigezogen werden (da man in der Schrift die vereindeutigende Satzbetonung ja nicht erkennen kann):

(13) *Die Mutter hat Maria angerufen, (den Vater Fritz).*

Solche Umstände lassen es als fraglich erscheinen, ob für den speziellen Fall der Satzgliedfolge im Mittelfeld die einheitliche Verarbeitung variabler Satzgliedfolge anwendbar gemacht werden könnte. Auf jeden Fall spielt diese Möglichkeit erst bei der gleichzeitigen Lösung der damit verknüpften semantischen Probleme eine Rolle.

3.7. Ausgeklammerte Probleme

Mit Absicht wurden in den Erörterungen dieses Abschnitts Probleme ganz ausgeklammert, deren Behandlung schon von der Theorie her ein Problem ist und deren Lösung, die einer Theorie adäquat wäre, jenseits der Möglichkeiten praktisch verwendbarer Systeme liegen würde.

Das wichtigste dieser Probleme stellt die Verarbeitung koordinierter Strukturen dar.

Es ist ja so, daß unter bestimmten inhaltlichen Bedingungen praktisch beliebige Ketten koordiniert werden können. Es ist prinzipiell nicht möglich, abschließend eine Liste von koordinierten Konstituenten anzugeben. Eher müßten wir Regeln zweiter Ordnung zur Verfügung haben: "Wenn X eine grammatische Struktur von bestimmten Eigenschaften ist, dann ist X' eine entsprechende koordinierte Struktur". Dies bedeutet, daß wir eine Grammatik haben müssen, in der solche Regeln zweiter Ordnung ausdrückbar sind, und daß wir dazu einen geeigneten Parser konstruieren müßten. Abgesehen davon, daß schon dies eine ziemlich komplexe Aufgabe darstellt, ist die Analyse einer koordinierten Struktur selbst wieder mit so vielen semantischen und pragmatischen Problemen verknüpft, daß ein funktionierender Apparat insgesamt recht utopisch wirkt. Stattdessen muß man sich wohl mit der Beschreibung ausgewählter und darstellbarer koordinierter Strukturen im Format üblicher Regeln begnügen.

Ähnliche Probleme wie koordinierte Strukturen stellen Komparativkonstruktionen dar. Der Komparationsmarke *als* können praktisch beliebige Teilketten vorangehen und folgen, die nicht einmal vollständige Konstituenten bilden müssen.

(14a) *Ich habe mehr gelbe als rote Rosen gekauft.*

(14b) *In Mannheim gibt es mehr Wohnungen mit als ohne Zentralheizung.*

Die Strukturen vor und nach *als* müssen sich jedoch in bestimmter Weise ergänzen. Auch um diese Regeln adäquat zu erfassen, würde man wohl Regeln zweiter Ordnung benötigen. Die Konstruktion eines Parsers, der solche Regeltypen verarbeiten kann, geht jedoch erst recht über praktikable Möglichkeiten hinaus.